Covering the TI99/4A, the Myarc 9640 and compatibles

MICROpendium

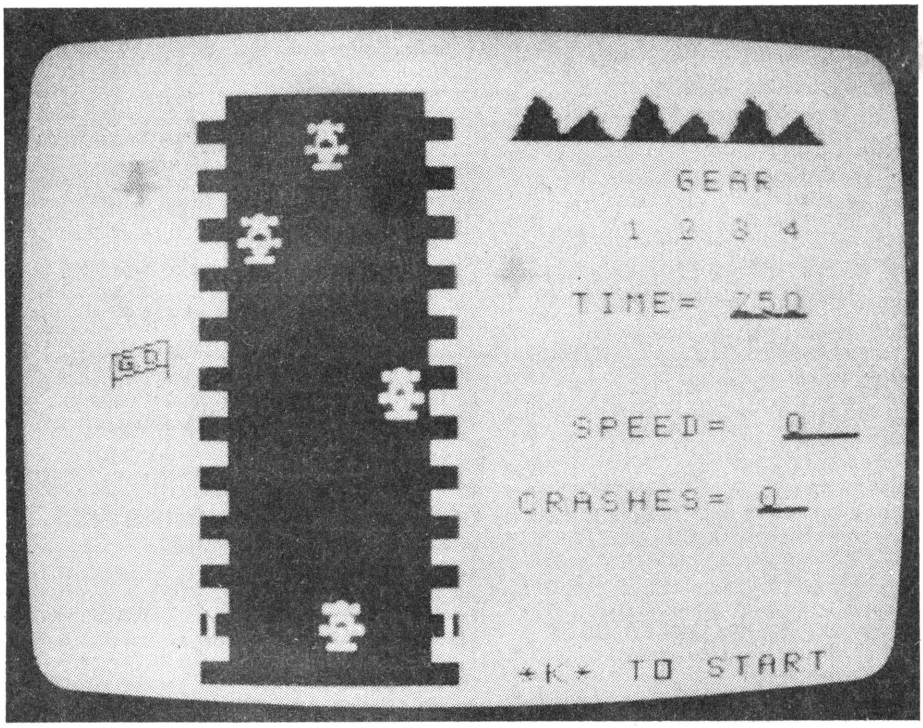
Volume 5 Number 7 August 1988 $2.00

MICRO-WORD

Part II of John Birdwell's assembly language word processor.

Wipe-out

An Extended BASIC game to test your reflexes in a race against the clock.



INSIDE

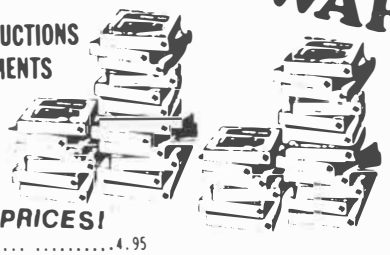
- A primer on Texas Instruments 9918A video chip
- News about upcoming Geneve software, and a program to set the Geneve's clock
- Reviews of the Gramulator and Barrage

The BIGGEST TI-99/4A SOFTWARE OFFER EVER

TEX-COMP™
TI USERS SUPPLY

PRESENTS:

TEXAS INSTRUMENTS & TEX-COMP Announce DRASTIC PRICE REDUCTIONS ON TEXAS INSTRUMENTS COMPUTER SOFTWARE



DISKETTE PROGRAMS **NEW LOW PRICES!**

PHD 5026	Bridge Bidding I	4.95
PHD 5039	Bridge Bidding II	4.95
PHD 5041	Bridge Bidding III	4.95
PHD 5020	Music Maker Demo (use with module)	4.95

CASSETTE PROGRAMS
see disk versions for req. i.e. TE-II

PHT 6009	Music Skills Trainer	4.95
PHT 6011	Computer Music Box	4.95
PHT 6018	Market Simulation	4.95
PHT 6031	Speak & Math	4.95
PHT 6042	Spell Writer	4.95
PHT 6026	Bridge Bidding I	4.95
PHT 6039	Bridge Bidding II	4.95
PHT 6041	Bridge Bidding III	4.95
PHT 6020	Music Maker Demo (use with module)	4.95

MANAGEMENT AND SMALL BUSINESS

MODULES

PHM 3006	Home Financial Decisions	4.95
PHM 3007	Household Budget Management	4.95
PHM 3022	Personal Real Estate	4.95
PHM 3016	Tax/Investment Rec. Keeping (disk req.)	4.95
PHM 3035	Terminal Emulator II	9.95
PHM 3044	Personal Report Generator (PRK req)	10.95
PHM 3113	Multiplan	24.95
PHM 3112	TI Writer	29.95

DISKETTE PROGRAMS **NEW LOW PRICE!**

PHD 5001	Mailing List (upgraded version)	4.95
PHD 5003	Personal Financial Aids	4.95
PHD 5021	Checkbook Manager	4.95
PHD 5022	Finance Manager	4.95
PHD 5024	Inventory Management	4.95
PHD 5027	Invoice Management	4.95
PHD 5029	Cash Management	4.95
PHD 5038	Lease/Purchase Decisions	4.95
PHD 5075	TI/Multiplan upgrade disk	4.95

CASSETTE PROGRAMS

PHT 6003	Personal Financial Aids	4.95
PHT 6038	Lease/Purchase Decisions	4.95

Send for 1988 Tex-Comp catalog & buyer's guide only $2.00 (comes with $5 savings certificate)

TI-COUNT SMALL BUSINESS SOFTWARE

General Ledger
Accounts Receivable
Accounts Payable
Inventory
Payroll
Mail System

SPECIAL 1988 OFFER--ALL SIX PROGRAMS PLUS
AUTO COUNT AUTO EXPENSE
RECORD PROGRAM..... ...$99.95

A $250.00 SAVINGS !!!!!!

MATH AND ENGINEERING

DISKETTE PROGRAMS **NEW LOW PRICE!**

PHD 5006	Math Routine Library	4.95
PHD 5008	Electrical Engineering Library	4.95
PHD 5013	Graphing Package	4.95
PHD 5016	Structural Engineering Library	4.95
PHD 5044	AC Circuit Analysis	4.95

CASSETTE PROGRAMS

PHT 6006	Math Routine Library	4.95
PHT 6008	Electrical Engineering Library	4.95
PHT 6013	Graphing Package	4.95
PHT 6016	Structural Engineering Library	4.95
PHT 6044	AC Circuit Analysis	4.95

****SPECIAL--ALL 5 OF THE ABOVE ON DISK OR TAPE---17.95

SPECIALS

Original TI Joysticks $7.95 (pair)

Replacement Console Power Supply (external transformer) $9.95

Replacement 99/4A Keyboards (plug in connection) $7.95

Cassette Cable $5.95

Console Dust Covers $7.95

PROTECT YOUR INVESTMENT WITH A BACK-UP TI99/4A

At only $79.95 the Texas Instruments 99/4A home computer is still the wisest choice for any individual or family just starting out in computing. But for those of you who already own a 99/4A and have purchased hundreds or even thousands of dollars in peripherals and software, buying a back-up computer for under eighty dollars is the smartest and least expensive way to protect the investment in your system. *$79.95**

TI-99/4A

Are you using your 99/4A in the office and wish you had another to use at home? Is your family squabbling over who gets the computer first? You can put your back-up computer to use at once to solve these problems, and rest assured that your primary system is protected too.

Time is running out. The Texas Instruments home computer will not be available for sale much longer. Buy your back-up TI-99/4A now and protect your home computer system investment for the years to come.

*Shipping, handling & insurance on this special offer is $10.00 (Continental U.S.) to any UPS deliverable address, HI, AK, Canada and APO slightly higher.

Send order and make checks payable to

TEX-COMP

P.O. BOX 33064 - GRANADA HILLS CA 91344

AUTHORIZED DEALER

VISA MasterCard

VISA and MASTERCARD HOLDERS CALL DIRECT
(818) 366-6631
24 Hour Order Line

TERMS: All prices F.O.B. Los Angeles. For fastest service use cashiers check or money order. Add 3% shipping and handling ($3.00 Minimum). East of Mississippi 4½%. Add 3% for Credit Card orders. Prices and availability subject to change without notice. We reserve the right to limit quantities.

NOTE: Payment in full must accompany all orders. Credit card, Company check or Money order for immediate shipment. Personal Checks require up to 4 weeks to clear. California orders add 6½% sales tax.

Contents

MICROpendium

MICROpendium is published 12 times annually for $20 per year by Burns-Koloen Communications Inc., 16606 Terrace Dr., Austin, TX 78728. Application to Mail at Second-Class Postage Rates is Pending at Round Rock, Texas. POSTMASTER: Send address changes to MICROpendium, P.O. Box 1343, Round Rock, TX 78680.

Display advertising deadlines and rates are available upon request.

All correspondence should be mailed to MICROpendium at P.O. Box 1343, Round Rock, TX 78680. We cannot take responsibility for unsolicited manuscripts but will give consideration to anything sent to the above address. Manuscripts will be returned only if a self-addressed stamped envelope is included.

Foreign subscriptions are $25.25 (Canada and Mexico); $23.50, surface mail to other countries; $37 airmail to other countries.

All editions of MICROpendium are mailed from the Round Rock (Texas) Post Office.

Mailing address: P.O. Box 1343, Round Rock TX 78680
Telephone: (512) 255-1512
Source: TI4596
CompuServe: 75156,3270
Delphi TI NET: MICROPENDIUM
GeNie: John Koloen

John Koloen.....Publisher
Laura Burns.....Editor

Programming conventions

Here are some tips to help you when entering programs from MICROpendium:

1. All BASIC and Extended BASIC programs are run through Checksum, the numbers that follow exclamation at the end of each program line. Do not enter these numbers or exclamation points. Checksum was published in the November 1987 edition.
2. Long XBASIC lines are entered by inputting until the screen stops accepting characters, pressing Enter, pressing FCTN REDO, cursoring to the end of the line and continuing input.

Best Buys From TENEX

No Gimmicks, No Hidden Charges, No Nonsense, Just Low Prices and Great Service!

FREE

Dust Cover & Everything Book

Discover the savings and easy shopping available from TENEX Computer Express PLUS receive a **FREE** dust cover for your TI! Cover is anti-static, 8-gauge vinyl sewn to our exacting standards with reinforced seams. Custom tailored with exclusive rear corner slit accommodating cables and speech synthesizer. Show your computer you care. Get to know our great products, extensive selection and fast service with a **FREE** copy of our *Everything Book for TI.* ($2.95 Shipping Charge)

10017 TI Console Cover & Catalog **(M6A**

From microFal

The 39¢ Diskette

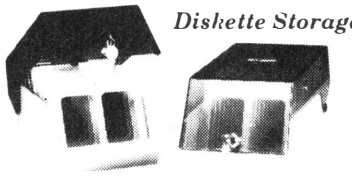
Are you paying too much for diskettes? Try our first quality, prime, 5-1/4" diskettes (no rejects, no seconds) at these fantastic sale prices and save, save, SAVE! Disks are packaged in boxes of 50; including diskettes in sleeves, labels, and write-protect tabs.

Each diskette is certified to be 100% error free and comes with a lifetime warranty (if you have a problem, we'll replace the diskette). All diskettes include hub reinforcement rings and write-protect notch.

All diskettes are double-density and work in either single or double-density drives.

SS, DD Diskettes, Box of 50
32391 **$19.50 - 39¢ ea.!**
DS, DD Diskettes, Box of 50
32403 **$24.50 - 49¢ ea.!**

Diskette Storage



- 100 disk (5-1/4") capacity.
- Lock and keys for extra security and easy carrying.
- Includes 8 index dividers with labels for organization of filing and retrieval.
- Made of durable anti-static, high impact plastic.
- Attractive smoke color lid. Sug. Retail $19.95

66826 **NOW ONLY $9.95**

Hardware

TI 99/4A Software

Printers

Okidata 180 Printer	$224.95
Okidata 120 Printer	$189.95
Okidata Microline 183	$279.95
Star NX-1000	$CALL
Star NX-1000 Rainbow	$CALL
Star NX-15	$CALL
Seikosha SP-180	$169.95
Seikosha SP 1200-AI	$199.95
Seikosha SP 1600-AI	$CALL

Computers

Geneve 9640	$CALL
Amiga 500	$CALL
Amiga 2000	$CALL
TENEX Turbo (Loaded)	$595.00

Monitors

13" Color Composite	$CALL
Magnavox CM8762, Color	$269.95

Joysticks/Controllers

NEW! Wico Ergostick	$19.95
Epyx 500XJ	$15.95
SuncomTac 5	$14.95
TI Adapter	$5.95

P-Box Cards

CorComp RS-232	$89.95
CorComp 32K	$119.95
CorComp 9900 Disk Cont.	$149.95
CorComp 512K	$253.00
Myarc RS-232	$99.95
Myarc 512K w/XB	$109.95
Rave Speech Card	$49.95

Accessories

Rave 105 Keyboard	$199.95
Universal Printer Stand+	$14.95
Data Transfer Switch	$24.95
RF Modulator (99/4A)	$19.95
Power Transformer (99/4A)	$19.95
1/2 Ht. 360K Floppy Drive	$99.95
Parallel Printer Cable 5 ft.	$24.95
Parallel Printer Cable 10 ft.	$34.95
Composite Monitor Cable	$7.95
TI Cart. Expander	$19.95

Ribbons

NX-1000 Black	$5.95
Okidata 120/180/183	$7.95
Axiom GP-550	$10.95
Gemini 10X,SG-10/15	$2.99
NX - 1000 Rainbow	9.95

Adv. Series Pkg., Disk	$17.95
Adv. Series Pkg., Cass.	$17.95
Better Banners, Disk	$19.95
Centipede, Cart.	$7.95
Certificate 99, Disk	$19.95
Console Writer, Cart.	$14.95
Donkey Kong, Cart.	$7.95
Font Writer II, Disk	$24.95
Nibbler, Disk	$9.95
Pac-Man, Cart.	$7.95
PEP, Disk	$59.95
Pro Typer, Cart	$19.95
Printers Apprentice, Disk	$19.95
Print Wizard, Disk	$19.95
Terminal Emulator II	$19.95
TI Artist, Disk	$19.95
TI Logo II (module)	$29.95
TI Planner, Cart.	$26.95
TI Planner Plus, Cart.	$49.95
QS Sideways, Disk	$14.95
Real TI/IBM Connection	$59.95
Word Writer, Cart.	$39.95
Word Writer Plus, Cart.	$59.95
4A Flyer, Cart	$19.95
4A Talk, Disk	$19.95
99 Fortran, Disk	$49.95

Free! 48 Page "Everything" Book With Any Order!

$89.95

Get TI-Count Accounting For Less

Pike Creek Accounting Package

This is the only accounting system to be endorsed for the TI 99/4A by Texas Instruments. It includes all the features of systems costing up to thousands of dollars for only a fraction of the price. You get all these: *General Ledger* - monitors up to 650 accounts, *Accounts Receivable* - keeps up to 200 customer accounts, *Accounts Payable* - features the ability to hold 200 vendor ledger accounts, *Inventory* - stores up to 700 items in 4 departments, *Payroll* - provides you with payroll checks and stubs of your choice, *Mail System* - change, move, or merge files. Get all 6 systems plus a bonus, the *Auto - Count Tax Log.*

75833 Pike Creek Accounting Package **$89.95**

$24.95

Give The "Foot" The Boot

You'll find our Peripheral Extension Cable to be a lifesaver. One end of this 30" cable plugs snugly into the expansion port on the right side of the 99/4A console; the other connects to the 44 contact slot of the PE Box cable, or any other stand-alone peripheral (such as the speech synthesizer, memory expansion, etc.). Customize your system set-up today!

22633 Peripheral Extension Cable **$24.95**

THE BEST PRICES
THE BEST SERVICE

WHY SHOP ANYWHERE ELSE?

Gives You More Control

PIO PRINTER INTERFACE PLUS. You'll be able to connect any parallel input printer directly to your computer. Just plug one end into the side of your console and the other into the printer. 120 day warranty. From CorComp. Sug. Retail $59.95

42250 Parallel Printer Interface **$49.95**

***Only* $49.95**

From Your Friends At

TENEX Computer Express

We gladly accept mail orders!
P.O. Box 6578
South Bend, IN 46660

Questions?
Call 219/259-7051

Ad M6C

Shipping Charges

Order Amount	Charge
less than $19.99	$3.75
$20.00-$39.99	4.75
$40.00-$74.99	5.75
$75.00-$149.99	6.75
$150.00-$299.99	7.75
$300.00 & up	8.75

MasterCard · American Express Card · VISA

No Extra Fee For Charges!

ORDER TOLL FREE
1-800-348-2778

WE VERIFY CHARGE CARD ADDRESSES

APO, FPO, AK, HI, CN, VI, GU, and foreign orders are subject to additional shipping charges. NOTE: Due to publishing lead times, product prices and specifications are subject to change without notice.

Comments

Background on the chips

There are a number of long, somewhat technical articles this month, including a piece on the 9918A VDP chip by Tony Lewis. While the information in the article isn't likely to send many readers into soaring flights of fancy, it is readable and provides a solid background on the inner workings of this small but significant piece of silicon. Next month Tony will be following with an article on the Yamaha 9938 VDP chip, which is used in the Geneve. We hope this provides some insight for 9640 users.

GENEVE DEVELOPMENTS

Several things:

• MDOS 1.06 has been released, as has GPL 1.02 and Myarc Disk Manager V Ver. 1.21. MDOS 1.06, running with GPL 1.02, allows users to automatically load programs from powerup. I find it to be very handy and a welcome feature. Also, the Print Screen function works.

• A timetable of software releases is published elsewhere, including Pascal, Advanced BASIC, MY-Pro-Art (an extensive rewriting of MY-Art), and GEME. I've seen an unfinished version of GEME (a windowing, mouse-driven, multi-tasking environment) and it looks impressive. Through the use of up to four user-defined windows, users can load and run four programs simultaneously and jump between them with a click of the mouse.

• MICROpendium will publish a question and answer column about the Geneve and other Myarc products (such as the hard and floppy disk controller card) starting next month. Anyone with questions they'd like us to ask is encouraged to drop us a line.

• Last month we published a list of software that readers most wanted to see running on the Geneve. Here is information about the software, provided by Myarc.

A number of programs that were mentioned by readers already work on the Geneve, including Plato, Logo II, Personal Record Keeping, Disk Managers I, II and III, Donkey Kong (run at speed 2), Submarine Commander and River Rescue. Geneve users who have not been able to get these cartridges to run on their 9640s are advised to send Myarc a good copy of the non-working program on disk. Send it to Myarc Alabama office (2624 Ranier Dr., Birmingham, AL 35215).

While readers want to see Q*Bert running on the Geneve, as well as Ms. Pacman and other Atari-type cartridges, the company is legally restricted from modifying the cartridges. Non-cartridge software, such as 99/4A Spell Check, like the Atari cartridges, is copyrighted and Myarc can't do anything to modify it without infringing on the copyright.

UPCOMING SOFTWARE

I expect to see a number of new programs being released for the TI and the Geneve between now and November, possibly including a word processor like no other in use on the TI.

HARD AND FLOPPY DISK CONTROLLER

Myarc shipped hundreds of HFDC cards in August. At this point, the card is used as a hard-disk controller only the Geneve. On the TI, it serves as both a hard and floppy disk controller.

One word of caution, which comes from experience: be careful when pulling the thing out of your PEB. Apparently, if the card is pulled out haphazardly, it can short out against the back of the PEB. I recommmend that if you take the card out, have the PEB powered-down for five minutes and then, very carefully lift the card out absolutely straight so that the portion of it that sticks out of the computer doesn't brush against the PEB. Use two hands to do it. This advice comes after having blown the card out through carelessness.

Next month I'll detail my experiences in setting up a hard-disk system.

—JK

Reviewed in MICROpendium

1984

February: B-1 Nuclear Bomber, Tandon TM-100 Disk Drive, Void, Beanstalk Adventure, Microsurgeon, On Gaming, Database 500.
March: Star Trek, Escape From Balthazar, Garkon's Getaway, Sky Diver, Mail-Call, Prowriter 8510 Printer.
April: Monthly Budget$ Master, Budget Master, Home Budget, Thief, Donkey Kong, Khe Sanh.
May: Companion Word Processor, Q*Bert, Mad-Dog I & II, Programs for the TI Home Computer.
June: Creative Expressions Accounts Receivable/Accounts Payable, CDC 9409 Disk Drive, StarshipConcord, Lost Treasure of the Aztec, ASW Tactics II.
July: Theon Raiders, Introduction to Assembly Language for the TI Home Computer, Game of Wit, Pole Position
August: TE-1200, Tower, Galactic Battle, Galaxy
September: Wycove Forth, 99/4 Auto Spell-Check, QUICKCOPYer, Wizard's Dominion, Anchor Automation Mk XII Modem
October: Killer Caterpillar, ZORK I, Defender
November: 9900 Disk Controller Card/Manager, Super Bugger, Transtar 120S printer, Floppy-Copy, Data Base-X
December: Gravity Master, Data Base Manager System, Learning 99/4A Assembly Language Programming

1985

January: Super Sketch, Foundation Computing 128K Card, PTERM-99, TI-Runner
February: Super Extended BASIC, Beginning Assembly Language for the TI, ZORK II
March: Morning Star Software CP/M Card, WDS/100 Winchester Disk Drive, Sketch Mate, BMC Color Monitor
April: 9900 Micro Expansion System, Disk+Aid, Gemini 10X-15X
May: Character Sets and Graphics Design, Draw 'N Plot
June: GRAPHX, DATA BASE I
July: Acorn 99, Advanced Diagnostics
August: Model Dow-4 Gazelle, TI-Artist, PC-KEYS, Not-Polyoptics' Bankroll
September: Midnite Mason, Myarc 32K/128K Card, GRAPHX Companion
October: 4A/TALK, Extended BASIC II Plus, XB Detective, Console Writer 2.a
November: Foundation Z80A/80-column cards, 9900BASIC, Adventure Editor
December: Display Enhancement Package, Triple Tech

1986

January: BITMAC, Starcross
February: Night Mission, Peripheral Diagnostic Module, BA-Writer
March: Super Duper, Tunnels of Doom Editor, Business Graphs 99
April: U.S. Open Tennis, PRBASE
May: 4A Flyer, GRAM Kracker, Artist's Companion
June: Myarc Disk Controller Card, Maximem
July: Horizon RAMdisk, Old Dark Caves, Funlwriter, TI99/4A Macro Assembler
August: JOYPAINT 99, GPL Assembler, TI99/4A Intern, GPL Linker
September: Mechatronic 128K Card
October: TI-Forth Utilities, CorComp Memory Plus
November: Submarine Commander, PEP, MAX-RLE
December: GK Utility I and II and GRAM Packer, X-10 Powerhouse, RAVE 99/101

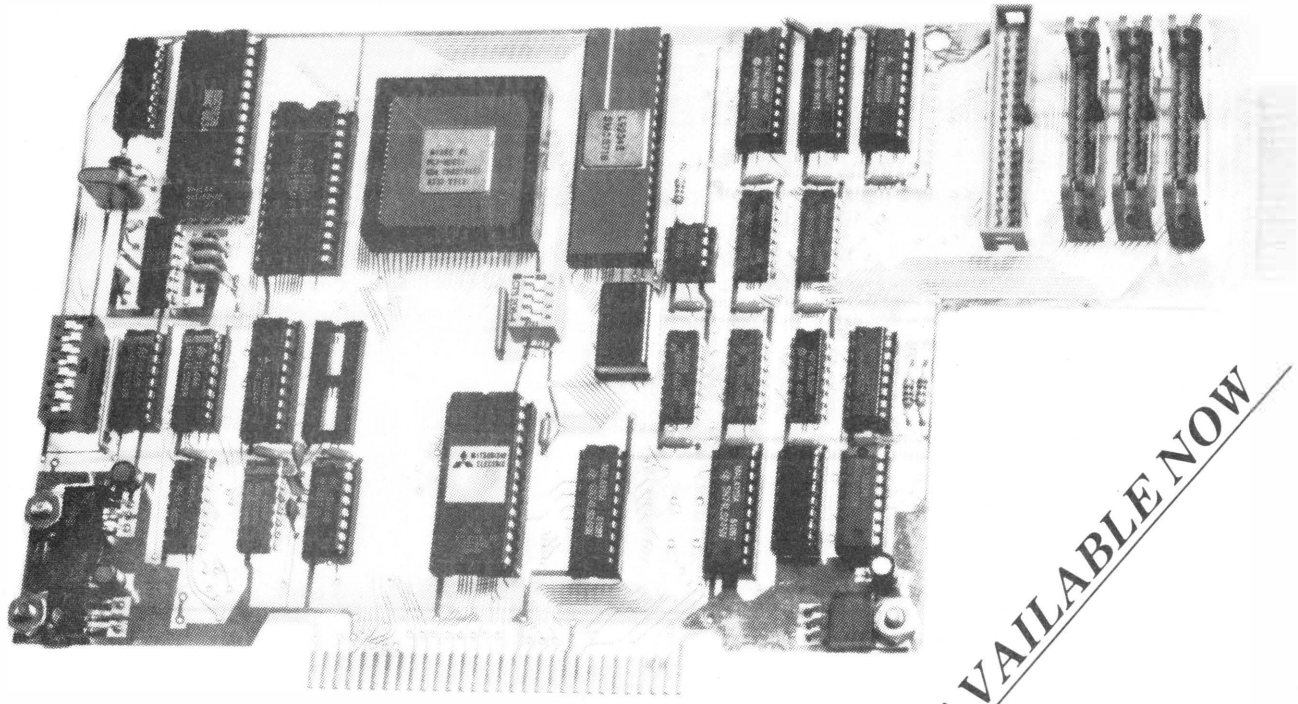
1987

January: MG DISKASSEMBLER, Myarc XBII
February: TI-Tax, Mechatronic Mouse
March: Wycove Forth version 3.0, DIJIT Systems RGB Conversion Kit, Spad XIII Flight Simulator
April: Geneve 9640, Disk Utilities
May: QS-Solitaire, Geneve 9640 (Part 2), Technical Drive, Console Calc
June: Character Sets and Graphic Design III, Writerease Ver. 1.1, 4A DOS, Prescan__It
July: Junkman Junior, Avatex 1200/1200hc modem, Bubble Plane
August: Prostick, The Brain, Rocketman, Menu Ver. 6.3
September: TI-IBM Connection, Super Extended BASIC
October: Fontwriter, Mechatronic 80-Column Card, Star NP-10 printer
November: Legends, Music Preprocessor, QS-Wheel, Spin-to-Win
December: Remind Me, Certificate 99, Myart-Art and Myarc Mouse

1988

January: Quik Font, EZ-Keys
February: Disk Utilities 4.0
March: Telco, String Master, Epson LX-800 printer
April: Super Space II, PC-Transfer, Calendar Maker, Archiver II Ver. 2.4
May: Plus!
June: Captain's Wheel 32K Memory Expansion, Desk Top Publisher Ver. 1.0, Texlink
July: Artist Enlarger

THE MYARC Hard & Floppy Disk Controller with Streamer Tape Backup Support



AVAILABLE NOW

Features include:

- Hard drive transfer rate of 5Mbit per second, for speed comparable to an external RAM disk card
- Interfaces with standard, off the shelf, hard, floppy and streamer tape drives
- Built-in real time clock, for time and date stamping of files
- Supports up to four 5 ¼" and/or 3 ½" floppy drives, mix or match
- All disk formats, SS/SD (90K), DS/DD (320/360K) and DS/QD (640/720K) supported
- MYARC Disk Manager V, the most intuitive and user friendly manager available
- One year limited warranty, 12 months parts, 6 months labor, is standard, an optional two, three or four year extended warranty is available

MYARC, Inc., P.O. Box 140, Basking Ridge, New Jersey 07920, Phone (205) 854-5843

Feedback

Don't be too hard on software makers

While I don't wish to be overly critical of the central point that Mr. Schmidt was attempting to make in his letter (June 1988), there are a couple of major factors that he needs to consider before coming down so hard on the producers of TI software.

(1) Vendors in the TI world have neither the budget nor the manpower of "Big Blue."

(2) Most vendors realize that even if they could produce what he is asking for, owners of an *orphaned* computer are not going to pay the prices required to offset time and expense.

The present situation is not likely to change in the near future. Our best hope is that producers will continue making software available to a dwindling number of users, and that, as a group, we fill in the missing pieces.

Ernie Pergrem
Palatine, Illinois

Machine routines obtained in UK

We (TI*MES, a United Kingdom users group) have now obtained — from US and Australian user groups — machine code routines for our unexpanded owners, who only need a 32K RAM to be able — using *cassette* — to merge programs from tape, and to save and load XB programs up to 22K. We also have the ability to supply them with many of the machine code programs we have (for instance we have a paid license to copy Micro Pinball for our members — a superb program!

One product in need of a review is the Bunyard manual, advertised for some months now. I gather a number of people have seen the ad and not ordered as they thought it was little more than a reprint of the TI technical manual. It is a *great* deal more.

A challenge: Using TI Base, emulate Personal Record Keeping plus Personal Report Generator. This should result in programmers becoming thoroughly at home using TI Base. It should give us a faster PRK!

And it will add a useful utility to the TI family.

Stephen Shaw
Stockport, Cheshire, England

Video Chess fix needed by user

Since subscribing to your magazine in January, I have found it to be an invaluable source of information on the TI and Geneve 9640. It is a relief to discover that the problems I've experienced with the Geneve were not unique and that Myarc, your staff and your readers are making significant progress to solving these problems.

Regarding the Geneve I have a couple of quick comments regarding the June issue. First, the list of TI programs that do not run as they should on the Geneve did not include Video Chess. Has a fix already been made, and if so, how could I order it? Second, while it is true that the fire button does not work on Pole Position, hitting the HOME key does what the fire button is supposed to.

Finally, would you be able to list vendors who either already do or are planning to carry Geneve software?

My continuing thanks to your staff for remembering that there are a large number of dedicated, contented computer users outside of the IBM-Apple-Commodore-Atari world.

Eric Wilson
Fremont, California

For you and other latecomers who didn't know, a fix to Video Chess was published in the November 1987* MICROpendium. *We are not sure of all the software that works with the 9640, nor do vendors always test that compatibility before releasing it, though we agree a list would be a good idea.— Ed.

Thank-you note

My TI99/4A experience has been one of friendship, caring and ongoing education from the very beginning as TIers autopiloted me from near and afar via letter directives as to its use and equipping.

While my TI and I "work" to help out in various ways at Marian Hall, it is informally and as a resident whose interest is deeply personal. After 40 years as a B.V.M. sister, the last eight years here due to a lung impairment, my commitment to "break shells" Ray Kazmer mentioned (Feedback, July '88) is an effort to add quality to the lives around me, to bring light and change into the darkness of confusion or simply a conversation starter in what could easily be a closed environment — be it through banners I color, music, or visual impact combined with sound. Being decades younger than most of our 135 residents, I sometimes see, hear and perceive differently.

Thank you each and every one who has helped me and enabled me to help others.

To name any person might be to miss one so please accept a general thank you to a fantastic network of caring, generous people.

Sister Pat Taylor, B.V.M.
Dubuque, Iowa

Program problems

Archiver II Vn. 2.4 will not work with my Geneve. I get a screen full of checkers for a split second, then the system locks up. I sent $10 to Barry Boone and he called me from school. I told him of the problem and he couldn't help.

Fast Term by J. Peter Hoddie runs very poorly and I have had to guess at some key presses to get by the second screen. A shame, the original FT ran great with the 99/4A. I spent five hours downloading archived files from The Source only to find that I couldn't unpack and run them (even on another TI). My system has the CorComp disk controller card and NEC Multisync Monitor.

Ali Ulgen
Seven Hills, Ohio

It's difficult to say what the problem with Archiver II is, in view of the fact that you've already contacted the author. We assume that it runs on a 99/4A and that the problem wouldn't be the result of a bad data bit. We have been running Ver. 2.4 for several months without a problem. Perhaps other readers might have a suggestion.—Ed.

Feedback is a reader forum. The editor may condense excessively lengthy submissions if necessary. We ask that writers limit themselves to one subject per submission. Our only requirement is that submissions be of interest to those using the TI99/4A or compatibles. Send items to MICROpendium: P.O. Box 1343, Round Rock, TX 78680.

from

TEX✦COMP

Serving the TI-99/4A user since 1980

THE BEST NAMES IN EDUCATIONAL COMPUTER SOFTWARE JOINED WITH TEXAS INSTRUMENTS TO PRODUCE A GREAT COLLECTION OF EDUCATIONAL PROGRAMS. TEX-COMP BRINGS THEM TO YOU AT A FRACTION OF THEIR ORIGINAL COST. NOW YOUR CHILDREN CAN LEARN FROM THE BEST FOR LESS.

SPECIALS

SAVE UP TO 85% OF ORIGINAL PRICES!!

TI Math Series

- ADDITION/SUBTRACTION I: Teaches basic arithmetic skills and provides drills for reinforcement. Grade Level 1. Speech synthesizer recommended. Cartridge.
- ADDITION/SUBTRACTION II: More difficult problems and techniques than found in Addition/Subtraction I. Grade levels 1-2. Speech Synthesizer recommended. Cartridge.
- MULTIPLICATION I: Teaches the basics of multiplication. Grade levels 3-4. Speech Synthesizer recommended. Cartridge.
- DIVISION I: Teaches the basics of division. Grade levels 3-5. Speech Synthesizer recommended. Cartridge.

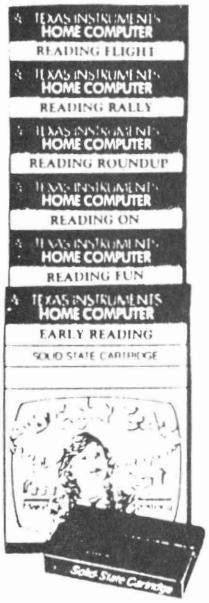
All 4 Modules for ONLY $19.95 + S&H

The Scott Foresman & Co. Reading Series with Special Value Package on the Three Most Popular Modules:

All 3 Modules for ONLY $14.95 + S&H

available separately at $9.95 each.

EARLY READING, READING FUN & READING ROUNDUP



Now you can purchase the entire Scott Foresman Reading Series at a fraction of the original price of over $325. This series consists of 6 graded modules and manuals which take the child from pre school through 6th grade level reading. This series has helped thousands of children to learn to read and to help improve their reading skills. By Special arrangement with Texas Instruments, **TEX-COMP**, is able to offer you a special package of the three most popular titles in this series, Early Reading, Reading Fun and Reading Roundup, at the amazing low price of only **$14.95** for all three (a 50 savings)!

- EARLY READING: Engaging color graphics and computer speech introduce and reinforce important reading skills. Speech synthesizer required. Cartridge.
- READING FUN: Stories and practice drills help build vocabulary and reading skills. Grades 1-3. Speech synthesizer is recommended. Cartridge.
- READING ROUNDUP: Several stories teach fourth grade level readers idioms, figures of speech and word meanings in context. Speech synthesizer optional. Cartridge.

Also available at $9.95 each are:

- READING ON: Teaches skills required to understand maps, graphs and schedules. Several stories and appropriate drills reinforce skills. Grade 3. Cartridge.
- READING RALLY: Practice basic reading skills and learn the difference between fact and opinion; understand the author's purpose and the bias and connotation of words. Grade 5. Cartridge.
- READING FLIGHT: For sixth grade level readers, several stories with special exercises to teach summarizing classifying and outlining skills. Speech synthesizer is optional. Cartridge.

The TI Speech Synthesizer is required with the Early Reading Module and recommended for several others.

SEE OUR EDUCATIONAL SOFTWARE MODULE SELECTION ON PAGE 3 OF THIS MAGAZINE!!

SEND ORDER AND MAKE CHECKS PAYABLE TO

TEX✦COMP™

P O BOX 33084 — GRANADA HILLS CA 91344

TERMS. All prices F O B. Los Angeles. For fastest service use cashiers check or money order. Add 3% shipping and handling ($3 00 minimum) East of Mississippi 4½%. (Free shipping on all software orders over $100 00). Prices and availability subject to change without notice. We reserve the right to limit quantities. When in the Los Angeles area visit our modern warehouse outlet store where you can purchase all TI items at our regular discount prices. Call for location & hours!

VISA MasterCard

VISA and MASTERCARD HOLDERS CALL DIRECT

(818) 366-6631

24 Hour Order Line

3% for credit card orders

BASIC

Some basics of BASIC

By REGENA

This month I'm going to discuss several of the fundamental BASIC commands that you might use in programming, plus answer some of the questions I have received recently.

You may have noticed program listings in which every line ends with an exclamation point and then a number. All BASIC programs in MICROpendium since the November 1987 issue have this format. When you are copying the program, *do not* type these extra characters on the line. The program has been processed through Tom Freeman's CHECKSUM program. These numbers help you enter a program correctly. When you type in a program using the Checksum program, you compare your checksum with the published number to make sure each line is correct. Refer to the October 1987 issue of MICROpendium for an explanation of the Checksum program. You may contact Tom Freeman through the LA 99ers Computer Group (P.O. Box 3547, Gardena, CA 90247-7247).

I can't seem to emphasize this next concept enough. If you are using a disk drive system and not using your program for file processing other than saving and loading your program, use CALL FILES(1) then press ENTER. Then type NEW and press ENTER. The screen will clear and the screen looks just like the beginning BASIC screen. Now you may load or type in a program and proceed normally. Many of my programs need this procedure to run without a MEMORY FULL error. I sue the CALL FILES(1) procedure as standard practice every time I sit down at my TI to program.

TI BASIC has several built-in features (other computers may need special utility programs) that you have available when you program. Use the command NUM (for NUMber) to number your lines automatically. This is equivalent to the commans AUTO on other computers. A line number appears at the left of the screen. Type in your BASIC statement, then press ENTER. Your line is entered and the next line number is automatically printed. To get out of the automatic numbering mode, simply press ENTER without typing anything else after the line number.

Notice that when you type NUM and press ENTER, the first line number is 100. Subsequent line numbers increment by 10. You may start the automatic line numbering anywhere by specifying a line number after NUM. For example, NUM 800 will start with Line 800, and NUM 57 will start with Line 57.

You may also specify the line increments by adding a comma and an increment value. The general form of this command is NUM n,i where n is the beginning line number and i is the increment. Here are some examples of this command.

Command	Line numbers
NUM	100, 110, 120, 130, . . .
NUM 500	500, 510, 520, 530, . . .
NUM 200,5	200, 205, 210, 215, . . .
NUM 650,50	650, 700, 750, 800, . . .
NUM 10,1	10, 11, 12, 13, . . .
NUM ,20	100, 120, 140, 160, . . .

By the way, the reason most of us number our program lines by 10s is so you can later add lines more easily if needed. TI BASIC does not allow multiple-statement lines, so to insert a statement between two existing statements, a new line number must be used. For example, if you have a Line 150 and a Line 160 and need to insert a statement between them, you may type 155 and the new statement. All TI BASIC statements must be numbered, and they will be automatically placed in the program in numerical order even if they are typed in a different order.

Another command very useful to programmers is RES (for RESequence). Other computers may use the command RENUM (for RENUMber). This command will make all the line numbers in your existing program sequential by a common increment. Any line numbers referred to in other statements will be automatically changed to the new numbering system. For example, if you have GOSUB 255 in your original programming and resequencing makes the old Line 255 become Line 320, the GOSUB 255 becomes GOSUB 320.

The general form for RES is RES n,i where n is the beginning line number for the first line of the program, and i is the increment between line numbers. This command works much like the NUM command.

Command	New line numbers
RES	100, 110, 120, 130, . . .
RES 200	200, 210, 220, 230, . . .
RES 300,5	300, 305, 310, 315, . . .
RES 500,50	500, 550, 600, 650 . . .
RES 1,1	1, 2, 3, 4, . . .
RES ,5	100, 105, 110, 115, . . .

I use NUM and RES quite often while I'm programming. For example, when I start programming, I type NUM to write the beginning lines of my program starting with Line 100 and incrementing by 10. I might start major sections of my program at Lines 1000, 2000 and 3000. Each time I start a new section I can use NUM n to start the automatic line numbers. For example, NUM 1000 will start the line numbers at 1000. I might start subroutines with easy-to-remember line numbers such as 500, 600, 700 and 800.

As I am programming and realize I have left out important commands, I can go back and insert lines. For example, if I have lines 210 and 220 and need to insert two statements, I can use Lines 214 and 217. What if I need to add more than nine lines between two existing statements. Use RES ,20 to increment lines by 20 or RES ,50 to increment lines by 50, and there will be plenty of possible line numbers between existing statements.

After the program is complete, I use RES to resequence all the statements and put thim in the somewhat standard form of starting with line 100 and incrementing by 10. This is how most of my programs are numbered for publication. The nice thing about using RES is that you can't tell where I planned poorly and had to add lines. All the lines are numbered nicely and you can use NUM to help you type in the program.

I have had several people wonder why I don't RESequence my published programs, because it is much easier to type in programs if you can use NUM. The answer is *I do!* If you see a copy that is not resequenced it may not be a final version, or it may be a version that someone else has customized by adding a few in-between lines.

I have had to RESequence a few programs in numbers other than RES 100,10. The reason is to save memory by using shorter line

(See Page 12)

Asgard Software is pleased to introduce:

PrEditor

A Programmer's Editor
By Tom Bentley

PrEditor is the ultimate text editor for the TI-99/4A or Myarc Geneve 9640! Completely written from scratch and extensively tested in a year of development, **PrEditor** is the perfect replacement for the Editor/Assembler or TI-Writer Editor. This incredibly fast program is loaded with features:

- **PrEditor** will allow you to have up to two files in memory at once. You can switch between them at a touch of a key.
- When switching between files the environment in each area is preserved - when you switch back and forth you are returned to exactly the point in either file that you were last at.
- You can cut and paste text between files - take pieces of one file and put it in another.
- You have a full range of text editing functions - insert, delete, and many others.
- **PrEditor** offers many block manipulation functions - move, copy, and delete blocks. Forget working with line numbers - simply mark the start and ending lines of the text you want to work with. All block operations are nearly instantaneous because of **PrEditor**'s unique design
- Includes full multi-directional search and replace commands
- Built-in disk cataloger included
- Has an "as-is" mode that lets you include in your file the exact ASCII character you type, from 1-255.
- You can view a file while editing two others
- Works on the Geneve in 80-columns or with any TI-99/4A 80-column card
- Will take full advantage of Geneve keyboard
- Allows you to use memory available in a Supercart, SuperSpace or Mini-Memory module for an even larger text buffer. 22K Buffer standard.
- **PrEditor** can be completely customized- you can permanently set for any copy the left and right margins, screen colors, the speed of the cursor in various operations, the number of columns it windows over (1-40 at a time) as you type, the tab stops, the video display available (40 or 80 columns), and finally, you can re-assign **ALL** the functions to any keys that you like. If you want to make the "roll-up" command **CTRL E**, you can!
- Finally, You can make as many copies as you like for personal use - and your personal needs. You can make custom copies for as many ways you use a text editor.

PrEditor is not only blazingly fast, it can be completely customized for any application - for writing articles, letters, or programs in c99, Fortran 99, Assembly and Extended BASIC. If you use the Editor/Assembler or TI-Writer editor and you are tired of resetting tabs (or not being able to do at all!), waiting for moves and copies, the screen windowing in big chunks, having to constantly load and save files to edit multiple files, and not being able to customize much of anything, then you *need* **PrEditor**!

$19.95 plus $.75 Shipping & Handling

Send to: **Asgard Software**
P.O. Box 10306
Rockville, MD 20850

Credit Card Orders: Call 1-800-456-9272. MC, Visa, AmEx accepted

REGENA ON BASIC—

(Continued from Page 10)

numbers — two digits instead of three digits, or even one digit instead of two or three. For example, "The Ugly Duckling" (MICROpendium, October 1987) is numbered by fives to save memory so it will fit on disk. Use NUM 10,5 to number your lines automatically (this is specified in the article). In San Francisco Revisited (MICROpendium, April 1988), the lines are numbered by ones because I really had to revise to get the TI99/4 cassette version to fit on disk for the TI99/4A. You can use NUM 1,1 to number your lines automatically as you type in this program.

One of the most used commands in BASIC is PRINT. You may print messages on the screen by using PRINT and then typing the message in quotation marks. If you have young children who want to learn to program, get them started by using the PRINT statement. The quote marks on the TI99/4A are typed by pressing the function key (FCTN, the key with the gray dot) and the letter P at the same time.

Our PRINT statement is quite versatile. PRINT without a message prints a blank line. To print a message, you use the quotes. A long message can be within the quotes, and the computer will print in the 28-character line. Use extra spaces to fit words on the screen properly.

You may put more than one message in the same PRINT command. Each message needs its own set of quotation marks, and you need to separate the messages with one or more print separators — a colon, a comma or a semicolon. A colon in the PRINT statement tells the computer to go to the next line. Several colons make the computer "go to the next line" several times. A semicolon tells the computer not to go to the next line but to print the next item right where the cursor is after the first item. A comma tells the computer to tabulate before printing the next message.

Remember that the print separators need to come after the quote mark ending the message. If the symbol is inside the quotes, it will be printed as part of the message.

Here is a short program to give you an idea how the PRINT statement can work.

```
100 REM PRINT
110 CALL CLEAR
120 PRINT "HELLO"
130 PRINT
140 PRINT "THIS IS A PRINTING EXAMPLE."
150 PRINT ::: "HERE ARE 3 COLONS"
160 PRINT "CHERY";" AND ";"BRAD"
170 PRINT "RICHARD","BOB","RANDY"
180 PRINT :"BRETT"
190 PRINT "LYNN"
200 END
```

You can experiment so you are familiar with what happens when you use commas, colons and semicolons after quotes or between messages. The separators can come after a message without another message in the same statement, and the next PRINT statement will be affected.

If your children are learning programming, they can try to print messages on the screen in a program, or they can use existing symbols to draw pictures. Have fun experimenting!

```
100 CALL CLEAR
110 PRINT "@@@@@@@@"
120 PRINT "@          @"
130 PRINT "@ 0        0 @"
140 PRINT "¦     ^     ¦"
150 PRINT "¦    \_/    ¦"
160 PRINT "  \         / "
170 PRINT "   \_____/ "
180 END
```

Demystifying assembly

Accessing peripheral devices

By JOHN BIRDWELL

I hope you have submitted your ideas on the features you would like to see incorporated into our word processing program so that we can soon begin to take a direction with it.

Before I begin discussing this month's topic I would like to correct an error in last month's publication. That error was in the copy list. You must also include the drive number of the source files, so the copy list would be as follows:

COPY "DSK1.MICRO-EQU" WHERE DSK1=YOUR DRIVE #

etc.

This month, while I await your input, I will be discussing the means by which access to a peripheral device is performed.

When TI designed the 99/4A they established a common method for accessing all external devices, this is the Device Service Routine Link or DSRLNK. Through the DSRLNK any manufacturer could easily add devices to our system as long as they adhered to the convention established for the DSRLNK. This has brought us such items as the Myarc and Corcomp disk controllers, Horizon RAMdisk, a variety of RS232/PIO cards, the Myarc 128/512K memory, the Myarc Hard/Floppy Disk Controller, and many others. Unlike other PCs, for the most part we have been able to simply plug a card into our PE boxes without concern about mixing hardware from different vendors. Talk to some of your friends who have expanded their IBM-PCs about the nightmares they have gone through attempting similar tasks and you can appreciate what TI did for us.

Access to devices in assembly is really not much different than access through BASIC. Before you can read from or write to a file you must OPEN and you must CLOSE it when done just like BASIC. This access is accomplished through the use of a Peripheral Access Block (PAB). In this PAB you must load all the information the DSRLNK needs to operate — OPEN, CLOSE, READ, WRITE or DELETE. The layout of the PAB is as follows.

BYTE	USE
0	Operation Code
1	File Type
2-3	VDP Buffer Address
4	Logical Record Length
5	Bytes Read or bytes to Write
6-7	Record Number for relative

(See Page 14)

ASSEMBLY—

(Continued from Page 12)

record fiLength for program image files

8 Used for cassette operations only

9 Length of Device Name

10 Device name

Operation codes are:

>00 OPEN
>01 CLOSE
>02 READ
>03 WRITE
>05 LOAD
>06 SAVE
>07 DELETE
>09 STATUS

File Type is done by bits within the byte. These are:

```
Byte 7 6 5 4 3 2 1 0
                  |— Relative file
                |— Write
              |— Read
            |— Internal
          |— Variable.
|————————— Used to provide return
```

status from DSRLNK.

For example, to read from a D/V file this byte would be have bits set to 00010100 in hex the byte would be a >14. To update this file you would have a >16 since you want to do both a read and a write.

VDP buffer is the address in VDP memory which the DSRLNK is to use for the READ/Write operations.

Logical Record Length is the maximum size of a record.

Your program must set the bytes to write for each record for Variable files and must be set to the Logical Record Length on Fixed files. On a read operation the DSRLNK will set the length of the record read for variable files so you should read this byte to determine the length of the record. On a Fixed length file this will naturally be equal to the Logical Record Length for Reads.

An example of a complete PAB to OPEN a D/V 80 file for a WRITE with the file name of DSK1.MICRO-DATA the PAB would be structured as follows:

DATA >0012, >1000, >5000, >0000, >000F
TEXT 'DSK1.MICRO-DATA'

After you have opened the file you would simply change the first byte, operation code, to a >03 to indicate a write operation. Then change it to a >01 to close the file.

Operation codes >05 and >06, LOAD and SAVE, are used with program image files. The DSRLNK will either load into or read from VDP memory beginning with the address you specified for the length given in bytes 6 and 7 of the PAB or until the end of file on a LOAD operation. This is the method used by the SAVE utility to create a program image file.

Basically its operation is to move a block of memory, beginning with the SLOAD/SFIRST address provided in the DEFs of your object code, D/F 80 file and then a SAVE PAB is passed to the DSRLNK. This continues until the SLAST address is reached. The maximum block size is 8192 which is actually 8186 bytes of data since a 6 byte header is added at the start of the file. This header is used by the E/A option 5 loader to determine where to place the data in CPU memory, the length of the data, and whether there are any additional files in the program. The format of the header is:

BYTES	USE
0-1	>0000 indicates no more files to load.
	>FFFF indicates more files to load.
2-3	Length of the file including the 6 byte header.
4-5	Start CPU address for the data and on the first file this is also the program entry point, which is why SLOAD and SFIRST must be the same.

This PAB is placed into VDP memory and its location is passed to the DSRLNK in CPU location >8356. What is actually placed into >8356 is the VDP location of the length of the Device Name within the PAB or PAB+9.

In our MICRO-WORD program we will be adding the ability to load a CHARA1, character definition, file. This will provide you with an example of this operation. Soon we will be adding the ability to read and write files so this will clarify this fully.

Also this month, we will be adding in a tab line and margins.

MICRO-WORD

```
********************************************************************
* To your MICRO-EQU add the following lines immediately after the *
*      B    @INIT line.                                            *
********************************************************************
*
* PLACE ITEMS WHICH WE WILL LATER DEVELOP CODE TO ENABLE US TO MODIFY AND
* SAVE DEFAULTS AT THE START OF THE PROGRAM SO IT WILL BE EASY TO CHANGE THEM
*
*
CHARPB DATA >0500            * LOAD
* SINCE CHARACTER SET STARTS AT >0800 MUST SKIP THE 6 BYTE HEADER WHICH
* PROCEEDS THE DATA IN A PROGRAM IMAGE FILE
       DATA >07FA            * ALLOW FOR 6 BYTE HEADER
       DATA 0                * NOT USED
       DATA >0400            * ALLOW FOR A FILE LENGTH OF 2048 BYTES
       DATA 11               * LENGTH OF THE CHARA1 FILE NAME
       TEXT 'DSK1.CHARA1'    * DISK & PLUS FILE NAME
*
* A DEFAULT TAB TABLE IN THE IMAGE OF TI-WRITER
* ALL TABS ARE >86 GREATER THAN THEIR POSITION RELATIVE TO 0
*
TABS   BYTE >80              * >80 = MARGIN  (LEFT)
       BYTE >8C              * >86 = LEFT MARGIN OF 0 THIS WILL BE COLUMN 6
       BYTE >80              * NOW RIGHT MARGIN
       BYTE >D1              * RIGHT MARGIN AT 75 * MAKE LESS THAN 27 FOR 4A
       BYTE >86              * NOW TABS STARTING WITH LEFT MARGIN
       BYTE >8A              * LETS SPACE THEM 10 APART STARTING A COL 5
       BYTE >94              * A MAXIMUM OF 16 TABS
       BYTE >9E
       BYTE >A8
       BYTE >B2
       BYTE >BC
       BYTE >C6
       BYTE >D0
       BYTE >D5              * SET UNUSED TABS TO CLOUMN 80
       BYTE >D5
       BYTE >D5
       BYTE >D5
       BYTE >D5
       BYTE >D5
```

(See Page 16)

MICRO-WORD—

(Continued from Page 14)

```
      BYTE >D5
      BYTE >80              * MARGIN AGAIN
      BYTE >86              * BACK TO COLUMN 0
*
* add the following EQUATE
IOVDP  EQU  >0600           * VDP ADDRESS FOR I/O PAB
*
* add the following DATA labels after
* ONELIN DATA 0
LMAR   DATA 0               * LEFT MARGIN
LMARTL DATA 0               * LEFT MARGIN IN TAB LINE
RMAR   DATA 0               * RIGHT MARGIN
RMARTL DATA 0               * RIGHT MARGIN IN TAB LINE
CURPOS DATA 0               * CURSOR POSITION ON TAB LINE
TLLIN  DATA 0               * START OF TAB LINE ADDRESS
*
* after the CURDEF statement add the following BYTE SAVE SEGMENT
* the  BSS reserves the amount of bytes specified
*
* TABLINE IMAGE
TABLIN BSS  80              * IMAGE FOR TAB LINE 80 COLUMNS LONG
*
* add the following after the statement
* FF    BYTE >FF
TABOFF BYTE >86             * VALUE OF TABS COLUMN ZERO
*
*********************************************************
* from MICRO-INIT file delete ALL line after        *
*      LWPI MAINWS            * SET BACK TO MAIN WS  *
* and add the following                              *
*********************************************************
*
      BL   @CLS             * GO CLEAR THE SCREEN AND RETURN
*
*****************************************
* WE WILL NOW LOAD THE USERS CHARA1 FILE IF IT IS PRESENT IF NOT JUST
* USE CHARACTER SET ALREADY DEFINED
*
      LI   R0,IOVDP         * MOVE CHARA1 PAB TO VDP
      LI   R1,CHARPB        * PAB DATA FOR CHARACTER SET LOAD
      LI   R2,21            * LENGTH OF PAB
      BLWP @VMBW            *
      LI   R0,IOVDP+9       * MUST TELL DSRLNK WHERE THE NAME LENGTH IS
      MOV  R0,@>8356        * MOVE TO DSR'S PAD LOCATION
      BLWP @DSRLNK          * NOW TO THE LOAD OF CHARA1 TO VDP
      DATA 8                * HIGH LEVEL DSR LINK
* NO NEED TO TEST IF CHARA1 GOT LOADED SINCE AND ERROR WILL NOT AFFECT
* WHAT WAS ALREADY THERE
*
* DEFINE THE CURSOR CHARACTER >0
*
      LI   R0,>0800         * START OF CHARACTER PATTERN TABLE
      LI   R1,CURDEF
      LI   R2,8
      BLWP @VMBW            * CURSOR CHARACTER
*
* Now we will invert the character set to provide inverse video when needed
*
      LI   R0,>0800         * Start of character set
      LI   R2,128*8         * The number of characters to do X 8 bytes/char.
      LI   R3,>0400         * Offset for inverted character set
CINVRT BLWP @VSBR           * Get a character
* the instruction INV will reverse all 1's and 0's
      INV  R1               * Invert the bits
* XOR exclusive OR will set any bits in the target register to a 1 unless
* they are already a 1 then it will set them to a 0
* In this case R3 with contains a >0400 will and its value will be toggled
* on and off as we read a character and turn it back off
      XOR  R3,R0            * Inverted set
      BLWP @VSBW            * Write out inverted bits
      XOR  R3,R0            * Back to main set
      INC  R0               * NEXT BYTE
      DEC  R2               * DONE ALL CHARACTER YET
      JNE  CINVRT           * NO
* NOW BUILD THE TAB LINE AND DISPLAY IT
      BL   @TABSET          * GO SET IT UP
      MOV  @ROW,R0          * NOW SET CURSOR POSITION TO LEFT MARGIN
      A    @LMAR,R0         * ADD IN LEFT MARGIN
      MOV  R0,@CURPOS       * SET INITIAL CURSOR POSITION ON TAB LINE
*
      MOV  @LMAR,R6         * SET R6 TO INITIAL DISPLAY POSITION

      B    @MAIN            * INITIALIZATION COMPLETE BEGIN THE PROGRAM
*
*****************************************************************
* in the file MICRO-MAIN remove the label MAIN                  *
* from the beginning of the file DO NOT REMOVE the BL @GETKY    *
*****************************************************************
*
* FIRST SHOW THE CURRENT CURSOR POSITION ON THE TAB LINE
MAIN   MOV  @CURPOS,R0      * GET CURSOR POSITION
       BLWP @VSBR           * GET THE CHARACTER THERE
       ANDI R1,>7F00        * TURN OFF THE INVERSE
       BLWP @VSBW           * WRITE IT BACK
*
*****************************************************************
* ADD THE FOLLOWING LINES TO THE END OF THE MICRO-DISP FILE     *
*****************************************************************
*
TABSET LI   R0,TABLIN       * START OF TAB LINE IMAGE
       LI   R1,'..'+>8080   * MAKE LINE INTO DOTS IN INVERSE VIDEO
       LI   R2,80           * 80 COLUMNS LONG
TABST1 MOV  R1,*R0+         * 2 DOTS
       DECT R2              * LOOP UNTIL ALL 80 COLUMNS ARE DOTS
       JNE  TABST1          * IF NOT ZERO CONTINUE
       LI   R0,TABLIN+9     * POSITION OF COLUMN 10
       LI   R1,'1 '+>8000   * NOW PLACE INDICATORS AT EVERY 10 COLUMNS
       LI   R2,8            * LOOP 8 TIMES
TABST2 MOVB R1,*R0          * MOVE IT
       AI   R0,10           * NEXT POSITION
       AI   R1,>0100        * MAKE DISPLAY 1 GREATER
       DEC  R2              * DONE ALL 8
       JNE  TABST2          * NO
* NOW FILL IN TABS AND MARGINS
       LI   R0,TABLIN       * START OF TABLINE
       LI   R1,TABS+4       * START OF TABS
       LI   R2,'T '+>8000   * TAB CHARACTER
       LI   R4,16           * MAXIMUM OF 16 TABS
TABST3 MOVB *R1+,R3         * TAB COLUMN
       SB   @TABOFF,R3      * SUBTRACT TABS VALUE
       SRL  R3,8            * MAKE A COUNT
       A    R0,R3           * POINT TO CORRECT COLUMN
       MOVB R2,*R3          * PUT IN A 'T'
       DEC  R4              * DONE ALL 16
       JNE  TABST3          * NO
       LI   R1,TABS+1       * POINT TO LEFT MARGIN
       LI   R2,'LR'+>8000   * LEFT AND RIGHT MARGIN CHARACTERS
       MOVB *R1+,R3         * COLUMN FOR LEFT MARGIN
       SB   @TABOFF,R3      * SUBTRACT THE VALUE FOR TABS
       SRL  R3,8            * MAKE A COUNT
       MOV  R3,@LMAR        * SAVE LEFT MARGIN VALUE
       A    R0,R3           * NOW POINT TO ITS COLUMN
       MOVB R2,*R3          * PUT IN THE 'L'
       INC  R1              * NOW POINT TO RIGHT MARGIN
```

(See Page 17)

MICRO-WORD—

(Continued from Page 16)

```
      MOVB *R14+,R3          * GET RIGHT MARGIN LOCATION
      SB   @TABOFF,R3
      SRL  R3,8
      MOV  R3,@RMAR          * SAVE RIGHT MARGIN
      A    R0,R3             * NOW POINTING TO RIGHT MARGIN LOCATION
      SWPB R2                * GET 'R'
      MOVB R2,*R3            * PUT IN THE 'R'
      LI   R0,22             * DISPLAY TAB LINE AT LINE 22
      MPY  @ONELIN,R0        * GET TO CORRECT VDP ADDRESS FOR SYSTEM TYPE
      MOV  R1,R0             * THE RESULT WAS IN R1 DO MOVE IT TO R0
      A    @LMAR,R1          * FIGURE TAB LINE LEFT MARGIN FOR DISPLAY
      MOV  R1,@LMARTL        * SAVE IT
      MOV  R0,R1             * NOW DO RIGHT MARGIN
      A    @RMAR,R1
      MOV  R1,@RMARTL        * SAVE DISPLAY TAB LINE RIGHT MARGIN
      LI   R1,TABLIN         * DISPLAY TAB LINE
      MOV  @ONELIN,R2        * SHOW THE AMOUNT FOR SYSTEM TYPE
      BLWP @VMBW
      RT                     * RETURN TO CALLER
*
*******************************************************
* IN THE FILE MICRO-KEYS MAKE THE FOLLOWING CHANGES *
*******************************************************
*
* change the NEWKY1 code as follows
*
NEWKY1 CLR  @RTIME           * RESET REPEAT TIMER
       MOVB R2,R1            * RESTORE THE ORIGINAL CHARACTER
       BLWP @VSBW            * DISPLAY IT
       MOV  @CURPOS,R0       * INVERT TAB LINE CHARACTER
       BLWP @VSBR            *
       AI   R1,>8000         * PLACE INTO INVERTED CHARACTER SET
       BLWP @VSBW            * WRITE IT BACK
       RT                    * RETURN
*
* change the ENTKY2 code as follows
*
ENTKY2 A    @ONELIN,R7       * START OF NEXT LINE
       MOV  R7,R6            * PLACE WHERE NEEDED
       A    @LMAR,R6         * PLUS LEFT MARGIN
* RESET TABLIN TO LEFT MARGIN
       MOV  @LMAR,R0         * LEFT MARGIN
       A    @EOE,R0          * ADD DISPLAY POSITION FOR TAB LINE START
       MOV  R0,@CURPOS       * MAKE THIS CURSOR TAB LINE POSITION
*
* change SCROLL CODE as follows
*
SCROLL MOV  @LMAR,R6         * FOR NOW JUST RESET TO TOP OF SCREEN
       B    @MAIN            * THIS WILL BE ADDED LATER
*
* replace the BACK and FOR code with the following
*
******** BACK SPACE *****************
BACK   C    R6,@LMAR         * AT THE LEFT MARGIN NOW
       JEQ  ENTKY4           * YES IGNORE
* MUST NOW BACKUP 1 ON TABLINE
* IF AT START OF TABLINE NOW GO TO RIGHT MARGIN
*   ELSE DECREMENT POSITION
       C    @LMARTL,@CURPOS  * AT LEFT MARGIN NOW
       JNE  BACK1            * NO
       MOV  @RMARTL,@CURPOS  * PLACE AT RIGHT MARGIN
* NOW GO UP A LINE ON DISPLAY
       S    @ONELIN,R6       *
* AND FIGURE THE START OF LINE POSITION
       BL   @BEGLIN          * MAKE R6 = TO START OF LINE
* NOW ADD RIGHT MARGIN TO THIS
       A    @RMAR,R6         * ALL SET
       JMP  ENTKY4           * LEAVE
BACK1  DEC  R6               * BACKUP DISPLAY
       DEC  @CURPOS          * AND TAB LINE
       JMP  ENTKY4           * LEAVE
*
********** FORWARD SPACE ***********
FOR    C    @RMARTL,@CURPOS  * AT RIGHT MARGIN NOW
       JNE  FOR1             * NO
       MOV  @LMARTL,@CURPOS  * SET TABLINE TO LEFT MARGIN
       A    @ONELIN,R6       * SET DISPLAY TO NEXT LINE
       BL   @BEGLIN          * POINT TO THE START OF THIS LINE
       A    @LMAR,R6         * ADD THE LEFT MARGIN VALUE
       JMP  ENTKY3           * GO TEST FOR END OF SCREEN
FOR1   INC  @CURPOS          * INCREMENT POSITION ON TAB LINE
       INC  R6               * AND DISPLAY
       JMP  ENTKY4           * DONE
*
* add the following to the end of the file
*
BEGLIN CLR  R5               * SETUP FOR DIVIDE
       DIV  @ONELIN,R5       * GET ABSOLUTE LINE # INTO R5 (0-23)
       MPY  @ONELIN,R5       * MAKE R6 = TO THE VDP LOCATION OF LINE
       RT                    * RETURN TO CALLER
*
* add the following to file MICRO-SRC
*      COPY "DSK1.MICRO-I/O"
*  N O T E :   THIS MUST BE INSERTED BEFORE THE FILE MICRO-END
*
***************** filename = MICRO-I/O
*
SCLEN  EQU  >8354
SCNAME EQU  >8356
CRULST EQU  >83D0
SADDR  EQU  >83D2
GPLWS  EQU  >83E0
SAVCRU DATA 0                * SAVED CRU
SAVENT DATA 0                * SAVED DSR ENTRY ADDRESS
SAVLEN DATA 0                * SAVED LENGTH
SAVPAB DATA 0                * SAVED PAB POINTER
SAVVER DATA 0                * SAVED DSR VESRION
NAMBUF DATA 0,0,0,0,0        * SAVED DEVICE NAME ie: DSK1
DLNKWS DATA 0,0,0,0,0        *DSRLNK WORKSPACE
TYPE   DATA 0,0,0,0,0,0,0,0,0,0,0,0
H20    EQU  $                * EQUAL BIT
H2000  DATA >2000
DECMAL TEXT '.'              * PERIOD
HAA    BYTE >AA              * FIRST BYTE OF A DSR
DSRLNK DATA DLNKWS,DLENTR
DLENTR MOV  *R14+,R5         * GET DSRLNK TYPE
       SZCB @H20,R15         * RESET CALLERS EQUAL BIT
       MOV  @SCNAME,R0       * GET THE POINTER TO PAB
       MOV  R0,R9            * SAVE IT
       AI   R9,-8            * POINT TO START OF PAB FILE TYPE
       BLWP @VSBR            * GET THE LENGTH OF FILE NAME
       MOVB R1,R3            * SAVE IT
       SRL  R3,8             * MAKE IT A WORD
       SETO R4               * INITALIZE COUNTER
       LI   R2,NAMBUF        * LOCATION TO SAVE NAME
LNK$LP INC  R0               * NEXT VDP LOCATION
       INC  R4               * ADD TO COUNT
       C    R4,R3            * READ THE WHOLE NAME
       JEQ  LNK$LN           * YES
       BLWP @VSBR            * GET A CHARACTER
       MOVB R1,*R2+          * MOVE IT TO NAMBUF
       CB   R1,@DECMAL       * GOT A PERIOD
       JNE  LNK$LP           * NOT YET GET SOME MORE
LNK$LN MOV  R4,R4            * ZERO LENGTH
```

(See Page 18)

MICRO-WORD—

(Continued from Page 17)

```
       JEQ  LNKERR          * YES ITS AN ERROR
       CI   R4,7            * IS NAME LENGTH >7
       JGT  LNKERR          * IF SO THIS IS AN ERROR
       CLR  @CRULST         * CLEAR CRU SAVE
       MOV  R4,@SCLEN       * SAVE THE LENGTH
       MOV  R4,@SAVLEN      * AGAIN
       INC  R4              * ADJUST IT
       A    R4,@SCNAME      * POINT TO END OF SAVED NAME
       MOV  @SCNAME,@SAVPAB * SAVE POINTER INTO NAME
SROM   LWPI GPLWS           * SET TO GPL WS
       CLR  R1              * CLEAR VSERION OF DSR
       LI   R12,>0F00       * START OF CRU-1 (>1000->1F00)
NOROM  MOV  R12,R12         * IS IT ZERO
       JEQ  NOOFF           * NO
       SBZ  0               * TURN OFF BOARD
NOOFF  AI   R12,>0100       * NEXT CRU
       CLR  @CRULST         * CLEAR SAVED CRU
       CI   R12,>2000       * REACHED LAST
       JEQ  NODSR           * YES DEVICE NOT FOUND
       MOV  R12,@CRULST     * SAVE CRU
       SBO  0               * TURN ON BOARD SO IT MEMORY MAPS TO >4000
       LI   R2,>4000        * START OF BOARD'S DSR CODE
       CB   *R2,@HAA        * IS IT VALID
       JNE  NOROM           * NO
       A    @TYPE,R2        * POINT TO TYPE OF DSRLNK
       JMP  SG02
SGO    MOV  @SADDR,R2       * CONTINUE
       SBO  0               * TURN ROM ON
SG02   MOV  *R2,R2          * GET NEXT ADDRESS VECTOR FOR TESTING
       JEQ  NOROM           * IF ZERO DEVICE NOT HERE
       MOV  R2,@SADDR       * SAVE ADDRESS
       INCT R2              * POINT ENTRY ADDRESS
       MOV  *R2+,R9         * SAVE IT FOR POSSIBLE ENTRY
       MOVB @SCLEN+1,R5     * GET LENGTH OF OUR DEVICE
       JEQ  NAME2           * ZERO LENGTH NO MATCH POSSIBLE
       CB   R5,*R2+         * IS LENGTH OF DSR'S DEVICE THE SAME
       JNE  SGO             * NO TRY ANOTHER
       SRL  R5,8            * MAKE A COUNT
       LI   R6,NAMBUF       * START OF OUR SAVED NAME
NAME1  CB   *R6+,*R2+       * COMPARE IT TO DSR'S
       JNE  SGO             * IF NOT EQUAL TRY ANOTHER
       DEC  R5              * ALL TESTED
       JNE  NAME1           * NOT YET
NAME2  INC  R1              * INC VERSION FOUND
       MOV  R1,@SAVVER      * SAVE IT
       MOV  R9,@SAVENT      * SAVE DSR ENTRY POINT
       MOV  R12,@SAVCRU     * SAVE CRU BASE FOR DSR
       BL   *R9             * GO EXECUTE DSR CODE
       JMP  SGO             * NOT RIGHT VERSION
       SBZ  0               * TURN OFF ROM
       LWPI DLNKWS          * BACK TO DSRLNK WS
       MOV  R9,R0           * POINT TO FILE TYPE BYTE IN VDP
       BLWP @VSBR           * READ IT
       SRL  R1,13           * SHIFT OUT EXTRA BITS
       JNE  IOERR           * IF ANY SET GOT AN ERROR
       RTWP                 * RETURN TO CALLERS WITHOUT EQUAL SET
NODSR  LWPI DLNKWS          * DSRLNK WS
LNKERR CLR  R1              * CLEAR ERROR FLAGS
IOERR  SWPB R1              * MOVE ERROR CODE TO TOP OF WORD
       MOVB R1,*R13         * SAVE THE ERROR CODE IN CALLERS R0
       SOCB @H20,R15        * SET EQUAL BIT TO INDICATE ERROR HAPPENED
       RTWP                 * RETURN TO CALLER
```

Trials of a c99 beginner

Mathematical function library

By CHARLES E. KIRKWOOD JR.

This article is the start of a Mathematical Function Library. The article didn't start out with that in mind, but one thing led to another until I thought that perhaps my attempted start might be useful.

The initial functions, with the results stored in r, are:

exp(x,r)	e to the x power
ln(x,r)	logarithm of x to the base e
ax(a,x,r)	a to the x power
an(a,n,r)	a to the integer n power
root(n,a,r)	nth root of a

The function **exp(x,r)** uses the series:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \ldots$$

The second function **ln(x,r)** uses the series:

$$\ln(x) = 2\left[s + \frac{s^3}{3} + \frac{s^5}{5} + \ldots\right],$$

where $s=(1-x)/(1+x)$.

The real variable **a** is raised to a power **x** by using logarithms. The function **ax(a,x,r)** takes the logarithm of variable **a** to the natural base **e** by using the function **ln()**, multiplies this by the power **x**, then converts this logarithm back to a number by the function **exp()**. The value of **a** cannot be negative. If **a** is zero, a test is made, and a value of zero is returned. The value of **x** can be positive, negative, or zero. Decimal powers can be used. Roots can be found by using values of **x** between zero and one, and reciprocals by negative values for **x**.

The function **an(a,n,r)** uses the same method as **ax(a,x,r)**, but if variable **a** is negative and exponent **n** is odd, variable **a** is changed to a positive value and the result is changed to a negative value. The value of **n** can be negative, zero, or positive. A value of zero is returned when incorrect arguments are used.

It is interesting to note that some of the problem solving methods devised by Newton and others before him are so applicable to the computer today. Some of the problems required a great deal of hand calculations and it is a wonder that they were ever solved.

One of the early problems is Newton's method for finding the root of a number. He used a method of averaging.

Let us first find the square root of a number. We all know that if R is the square root of A, that

$$R \times R = A \quad \text{or} \quad R = A / R$$

Newton guessed at the answer; let us call this guess RO. If the value of RO is less than the square root, then A/RO is greater than the square root, and if RO is greater than the square root then A/RO is less than the square root. Newton then said that a better answer could be found by taking the average of RO and A/RO, or

$$R = (RO + A/RO)/2$$

If R is not the square root, then set RO=R and repeat the process. Ideally, the correct solution is obtained when RO and A/RO are equal or when R equals the value of RO. This may not always be exactly possible with the digital computer.

As an example, let us take the square root of 5. My math table tells me the answer is 2.236068. Let the first guess be the number itself, 5.

RO	A/RO	RO+A/RO	(RO+A/RO)/2
5	1	6	3
3	1.666666667	4.666666667	2.333333333
2.333333333	2.142857143	4.476190476	2.238095238
2.238095238	2.234042553	4.472137791	2.236068896
2.236068896	2.236067059	4.472135955	2.236067978
2.236067978	2.236067978	4.472135955	2.236067978
2.236067978	2.236067978	4.472135955	2.236067978

In all the functions given, a satisfactory answer is assumed when the absolute value of two successive iterations divided by the last value is less than or equal to 0.000000001.

Newton modified this square root formula to take care of other roots and ended up with

$$R = \frac{(N-1)R0 + A/(RO)^{N-1}}{N}$$

A function **init()** is included to initialize the constants used in the functions. Since the real arithmetic is performed by functions, a BASIC subprogram is included to show the steps. I stored the functions in a file called MATH;C.

The following must be included with your program:

```
#include DSK1.FLOAT;C
#include DSK1.MATH;C
/*when an() is used also*/
#include DSK1.CONV;C
main()
{
your program
}
```

```
10 REM BASIC SUBPROGRAMS        /*THE MATH FUNCTIONS*/
                                /*LIBRARY*/

                                /*Global real variable*/
                                /*arrays used with the*/
                                /*Mathematical Functions*/

                                float mwone[8],zero[8];
                                float two[8],er[8],wone[8];

                                /*Function exp()*/
100 SUB EXP1(X,R)               exp(x,r)
                                float *x,*r;
```

(See Page 20)

c99—

(Continued from Page 19)

```
                          {
                            float ro[8],xo[8],ra[8];
                            float rd[8],k[8],i[8];
                            init();
110 K=1                     fcpy(wone,k);
120 RO=1                    fcpy(wone,ro);
130 XO=X                    fcpy(x,xo);
140 I=1                     fcpy(wone,i);
150 RD=1                    fcpy(wone,rd);
160 IF RD<=0.000000001      while(fcom(rd,">",er))
THEN 260                    {
170 K=K*I                     fexp(k,"*",i,k);
                              fexp(xo,"/",k,ra);
180 R=RO+XO/K                 fexp(ro,"+",ra,r);
190 XO=XO*X                   fexp(xo,"*",x,xo);
200 I=I+1                     fexp(i,"+",wone,i);
210 RD=R-RO                   fexp(r,"-",ro,rd);
220 IF R<>0 THEN RD=RD/R      if(fcom(r,"!=",zero))
                                fexp(rd,"/",r,rd);
230 IF RD<0 THEN RD=-RD       if(fcom(rd,"<",zero))
                                fexp(rd,"*",mwone,rd);
240 RO=R                      fcpy(r,ro);
250 GOTO 160                  }
                              return(r);
260 SUBEND                  }

                            /*Function ln()*/
300 SUB LN(X,R)             ln(x,r)
                            float *x,*r;
                            {
                              float ro[8],ra[8],rd[8];
                              float x1[8],x2[8],s[8],i[8];
                              float sq[8];
                              init();
310 IF X<=0 THEN R=0 ::       if(fcom(x,"<=",zero))
PRINT "IMPROPER ARGUMENT ",   {
"TO LN()" :: GOTO 470           fcpy(zero,r);
                                puts("Improper argu");
                                puts("ment to ln\n\n");
                              }
                              else
                              {
                                fexp(x,"-",wone,x1);
                                fexp(x,"+",wone,x2);
320 S=(X-1)/(X+1)               fexp(x1,"/",x2,s);
330 SQ=S*S                      fexp(s,"*",s,sq);
340 RD=1                        fcpy(wone,rd);
350 I=1                         fcpy(wone,i);
360 RO=0                        fcpy(zero,ro);
370 IF RD<=0.000000001          while(fcom(rd,">",er))
THEN 460                        {
                                  fexp(s,"/",i,ra);
380 R=RO+S/I                      fexp(ro,"+",ra,r);
390 S=S*SQ                        fexp(s,"*",sq,s);
400 I=I+2                         fexp(i,"+",two,i);
410 RD=R-RO                       fexp(r,"-",ro,rd);
420 IF R<>0 THEN RD=RD/R          if(fcom(r,"!=",zero))
                                    fexp(rd,"/",r,rd);
430 IF RD<0 THEN RD=-RD           if(fcom(rd,"<",zero))
                                    fexp(rd,"*",mwone,rd);
440 RO=R                          fcpy(r,ro);
450 GOTO 370                    }
                              }
460 R=2*R                     fexp(r,"*",two,r);
                              return(r);
470 SUBEND                  }

                            /*Function ax()*/
500 SUB AX(A,X,R)           ax(a,x,r)
                            float *a,*x,*r;
                            {
                              float z[8];
                              init();
510 IF A<0 THEN R=0 ::        if(fcom(a,"<",zero))
PRINT "IMPROPER ARGUMENT ",   {
"TO AX()" :: GOTO 580           fcpy(zero,r)
                                fpput("Improper argu");
                                fpput("ment to ax()\n\n");
                              }
520 IF A=0 THEN R=0 ::        else if(fcom(a,"==",zero))
GOTO 580                        fcpy(zero,r);
530 IF A=1 THEN R=1 ::        else if(fcom(a,"==",wone))
GOTO 580                        fcpy(wone,r);
540 IF X=0 THEN R=1 ::        else if(fcom(x,"==",zero))
GOTO 580                        fcpy(wone,r);
                              else
                              {
550 CALL LN(A,Z)                ln(a,z);
560 Z=Z*X                       fexp(z,"*",x,z);
570 CALL EXP1(Z,R)              exp(z,r);
                              }
                              return(r);
580 SUBEND                  }

                            /*Function an()*/
600 SUB AN(A,N,R)           an(a,n,r)
                            float *a,*r;
                            int n;
                            {
                              float an[8],z[8],x[8];
                              init();
610 AN=A                      fcpy(a,an);
620 IF A=0 THEN R=0 ::        if(fcom(a,"==",zero))
GOTO 700                        fcpy(zero,r);
630 IF A=1 THEN R=1 ::        else if(fcom(a,"==",wone))
GOTO 700                        fcpy(wone,r);
640 IF X=0 THEN R=1 ::        else if(fcom(x,"==",zero))
GOTO 700                        fcpy(wone,r);
                              else
                              {
```

(See Page 21)

c99—

(Continued from Page 20)

```
650 IF A<0 THEN AN=-AN
660 CALL LN(AN,Z)
670 Z=Z*X
680 CALL EXP1(Z,R)
690 IF(A<0)AND(INT(N/2)*2<>N)
THEN R=-R
700 SUBEND
800 SUB ROOT(N,A,R)
810 IF (N<>0)OR(INT(N/2)*2=N)
AND(A<0) THEN PRINT "THE ",
"ROOT CANNOT BE CALCULATED"
:: R=0 :: GOTO 970
820 AN=A
830 IF A=0 THEN R=0 ::
GOTO 970
840 IF A<0 THEN AN=-AN
850 M=N-1
860 RO=AN
870 RD=1
880 IF RD<=0.000000001
THEN 960
890 Z=1
900 FOR I=1 TO M :: Z=Z*RO
:: NEXT I
910 R=((N-1)*RO+AN/Z)/N
920 RD=(R-RO)/R
930 IF RD<0 THEN RD=-RD
940 RO=R
950 GOTO 880
960 IF A<0 THEN R=-R
970 SUBEND
```

```
if(fcom(a,"<",zero))
  fexp(an,"*",mwone,an);
ln(an,z);
itof(n,x);
fexp(z,"*",x,z);
exp(z,r);
if ((fcom(a,"<",zero))
&&((n/2)*2!=n))
  fexp(r,"*",mwone,r);
}
return(r);
}

/*Function root()*/
root(n,a,r)
int n;
float *a,*r;
{
  float an[8],ro[8],rn[8];
  float ra[8],rb[8],rd[8];
  float rn[8],z[8];
  int m,i;
  init();
  if((n<=0)|((n/2)*2==n)
  &&(fcom(a,"<",zero)))
   {
    puts("The root cannot ");
    puts("be calculated!\n");
    fcpy(zero,r);
    return(r);
  }
  else
  {
    fcpy(a,an);
    if(fcom(a,"==",zero))
    {
      fcpy(zero,r);
      return(r);
    }
    else
    {
      if(fcom(a,"<",zero))
        fexp(an,"*",mwone,an);
      m=n-1;
      fcpy(an,ro);
      itof(n,rn);
      fcpy(wone,rd);
      while(fcom(rd,">",er))
      {
        fcpy(wone,z);
        for(i=1;i<=m;++i)
          fexp(z,"*",ro,z);
        fexp(rn,"-",wone,ra);
        fexp(ra,"*",ro,ra);
        fexp(an,"/",z,rb);
        fexp(ra,"+",rb,ra);
        fexp(ra,"/",rn,r);
        fexp(r,"-",ro,rd);
        fexp(rd,"/",r,rd);
        if(fcom(rd,"<",zero))
          fexp(rd,"*",mwone,
          rd);
        fcpy(r,ro);
      }
      if(fcom(a,"<",zero))
        fexp(r,"*",mwone,r);
    }
  }
  return(r);
}

/*Function init()*/
init()
{
  int z,o,mo,th,h,t;
  float thou[8],hun[8];
  z=0;
  itof(z,zero);
  o=1;
  itof(o,wone);
  t=2;
  itof(t,two);
  mo=-1;
  itof(mo,mwone);
  th=1000;
  itof(th,thou);
  h=100;
  itof(h,hun);
  fexp(thou,"*",thou,er);
  fexp(er,"*",hun,er);
  fexp(wone,"/",er,er);
  return;
}
```

The algorithms used may not be the fastest, but they were the only ones I had. Several years ago I had a book which had some good mathematical algorithms for different mathematical functions. Well, I couldn't find this book, so for the first four, I used the algebraic series found in a mathematical handbook. For small numbers they work fine, but for large numbers they are slow. For now, they are better than nothing. You might have a better way yourself. If you do, I'd appreciate hearing from you. I searched through some books in the library but was unable to come up with any real fast algorithms. Will redo the functions if I find a better way.

Harms markets CLASS

Bill Harms is marketing CLASS for $10. The program is designed for use with the Super Extended BASIC cartridge. The program is used to create signs, posters and other graphics. CLASS may be ordered from Harms at 6527 Hayes Ct., Chino, CA 91710.

TEX COMP

FREEWARE DISKS

Public Domain and Shareware Programs
Are Available from our Library, and Priced at

Only $4.95 each

EXCITING NEW WAYS TO USE YOUR TI-99/4A COMPUTER

ORDERS SHIPPED OUT SAME OR NEXT DAY

GAMES

GREAT 99/4A GAMES VOL 1 (38) A collection of the very best. Professional quality.
GREAT 99/4A GAMES VOL 2 (39) Continuation of VOL 1 with more great action & graphics.
BEST OF BRITAIN VOL 1 (44) A collection of the best U.K. has to offer.
BEST OF BRITAIN VOL 2 (45) "Legend of Carfax Abby" an all graphics adventure.
GHOSTMAN (48) The fastest Pacman type game ever!
DEMON DESTROYER (49) Starts where Invaders leaves off.
OH MUMMY!! (50) Search the tomb for treasure while being chased by mummys.
BERLIN WALL (51) Excape from E. Berlin and avoid mines.
FREDDY (60) Great action and graphics. Excape from an underground cavern. Great!!
THE MINE (61) Fast action and great graphics. Hours of excitment with this one!
TI RUNNER II (70) An all new upgrade of one of the best!
CHESS (66) The famous game Zargon. Loads from extbasic.
CHECKERS & BACKGAMMON (33) A collection of the best.
SOLITAIRE & SCRABBLE (34) A classic game collection!
WHEEL OF FORTUNE, BLACKJACK & JOKER POKER (2) Three of the best we have seen. So good you will expect Vana to appear!
STRIP POKER (13) When you win she loses everything!
ASTROBLITZ/MAZOG (63) Two professional quality action games you are sure to like!
MAJOR TOM/SPACE STATION PHETA (64) Two great space games!
PERFECT PUSH (65) One of the finest games ever written. Fantastic action and top notch graphics. Space game!
SUPER TRIVIA 99 (46) The best trivia games we have seen. Complete with questions!
R RATED GAME DEMO (26) The classic Space Invaders with "unusual" guns & targets. For Adults Only!
TI-99 OLOPY' (12) Now you can play the famous board game right on your 99/4A. Do not pass GO!!!

EDUCATION

KIDS LEARNING VOL 1. (27) A 2-disk side collection of educational programs. Math, geography, reading and more.
MORSE CODE TRAINER (31) A professional program to learn and practice code.
ASTRONOMY (54) Plots the heavens and teaches you about the solar system.
KIDS LEARNING VOL 2. (71) Still more great learning programs. We only included the very best!

MUSIC

THE SINGING TI-99/4A (1) A 2 sided collection of songs where the computer actually sings. By Ken Gilliland. Requires speech syn.
TI MUSIC/GRAPHICS DEMO (5A) A great collection of music & matching graphics.
EXBASIC MUSIC DEMO (6) A 2-sided collection of great music with graphics. Hours of enjoyment!
COMPUTER PLAYER PIANO/CHORD ANALYSIS (69) A piano on the screen plays your selections or write your own with instructions incl. Also a program to learn keyboard chord formation.
EXBASIC XMAS MUSIC (32) A 2-disk side collection of christmas and holiday music. Completely menu selectable!

SPREADSHEETS

SPREADSHEET DEMO (56) A complete spreadsheet program for learning and many applications. Easy to learn and use!

ACCOUNTING AND FINANCE

ACCOUNTS RECEIVABLE (20) A complete AR program with documentation. Won 1st prize in TI programming contest.
STRICTLY BUSINESS (36) A 2-disk side collection of programs for evaluating loans, interest, stocks etc.

DATABASE PROGRAMS

DATA BASE DEMO (21) A fully set up data base program designed for filing and finding magazine articles. Easy to use or modify for other applications. Sample data included!
PR BASE (58) This is a full feature DB freeware program that is rated as one of if not the best. Documentation included!

GRAPHICS

ANIMATION 99' (52) This is the one by Ray Kazmer that was featured in the July 88 Micropendium. See fantastic animation and also learn how it was done. This one is destined to be a classic.
ANIMATED XMAS CARD (11) This is the original animation by Ray Kazmer that made him an overnight superstar in the TI community. This classic is also referred to as "Woodstock" among TI'ers.
PRINTART DEMO (4) This 2-disk side collection prints well known comic and TV personalities out on your printer.
FIGURE STUDY (14) This is a collection of programs that print Playboy type.center-folds out on your printer.
MONA LISA PRINTOUT (9) This program prints a near photo quality picture of Mona Lisa on your printer. You won't believe the quality!
SPACE SHUTTLE DEMO (7) An outstanding music/graphics program that salutes the U.S. space program. Its almost like watching a film.
STAR/EPSON DEMO (15) A 2-Disk side collection of programs to show you what your printer can really do. Also a great graphics tutorial with examples!
GOTHIC PRINT DISK (10) This program lets you type a message and then prints it out in Old English style. Looks like hand lettered calligraphy. Great for invitations, announcements.
SIDEWAYS PRINTOUT (16) Lets your printer print sideways. Great for spreadsheets and banners. Includes two versions and new Multiplan enhancements.
VIDEO GRAPHS (41) This disk is sold as a backup to owners of the discontinued TI Video Graphs module. We can only legally provide it to module owners.

TELECOMMUNICATIONS

TELCO (57) This program has been rated as one of the best telecommunications programs for the TI-99/4A. A user supported program that contains everything you need to upload and download data with your modem. Supports all baud rates and protocalls.

APPLICATIONS

WILL WRITER (23) Enter your answers to a group of questions and this program writes out a complete will.
MEDICAL ALERT (25) Contains many menu accessable files on what to do until the doctor or paramedics come. Could easily save a life!
ENGINEERING CALCULATIONS (24) A 2-disk side collection dozens of engineering and technical formulas. Does calculations, conversions, and even designs electrical circuits. Even contains medical and communications data and formulas.
LABEL MAKER (29) A pair of programs that let you make quick and easy labels for all purposes. Mail, disks, files etc. Uses standard tractor labels and even makes a graphic picture with the label text.
INFOCOM RAPID LOADER (47) A must for owners of Infocom 99/4A games. Loads games in seconds instead of minutes. Easy to use!
GENEALOGY (67) Now you can enter and arrange your family tree and print out copies for your relations. Also can be used if you breed animals such as dogs, cats or horses.
GRAPH MAKER (59) A collection of the best programs we have seen that produce graphs and charts from your data. Printer required!
HOUSEHOLD BUDGET PRINTOUT (30) This program lets you printout the data from the TI Household Budget module, an important feature that TI forgot.

Wipe-Out

Quick reactions will keep you in the driver's seat

By DAVID MENNENOH

Wipe-Out is a fast-paced racing game in which you must maneuver around three other cars on the course, being careful not to crash nor go off the road. The race continues until the on-screen timer runs out. Although you are given an unlimited number of cars to replace damaged ones, your final score is heavily dependent on the number of crashes you have.

If, when the timer runs out, you have had fewer than two crashes you will receive an extended play period in which you will travel even faster than before.

Your car is controlled by the K and L keys, which move the vehicle left and right, respectively. Shifting gears is accomplished by using the 1-4 keys. In gears 2-4 you move fast enough to pass other cars, but in first gear the other cars will pass you.

Scoring is determined by dividing the number of laps you have accumulated by the number of crashes. Thus, to get a high score, you have to travel fast and maneuver well.

Wipe-Out has approximately 8K of console memory left, a substantial amount. One thing you may wish to change initially is the key responsiveness. This, as is, is not bad, but may be increased significantly by removing the on-screen timer, as it takes a long time to continually update the on-screen time. You may also wish to change the scoring routine, because it scores a little on the low side.

So far, the highest score to date is 1,767. Happy motoring.

Program Explanation

10-60	Title screen
70-90	Clear screen, set game variables, set colors
100-200	Define the characters
210-270	Build the board
280	Delete sprites, check for extended play
290-320	Put the sprites on the screen
330-370	Display the game status
380	Screen color
390	K to start
400	Turn on the screen
410	Loop until K is pressed
420-430	Display underline, set speed to second gear
440-470	Start game, set sprites in motion
480	Decrement time, display time
490	Check for crash
500-570	Set keypress, move car
580	Switch gears
590-640	Car crashed, increment crash, crash sound
650	Crash, increment crash
660	Time up, check for extended play
670-760	Build game over screen
770	Display game over
780-800	Calculate score
810-830	Display scores, play again option
840-910	Play again?
920	Extended play routine

WIPE-OUT

```
10 CALL CLEAR :: CALL SCREEN
(2):: FOR T=1 TO 13 :: CALL
COLOR(T,16,2):: NEXT T !ØØ9
20 CALL COLOR(1,5,2,13,7,16)
:: CALL CHAR(128,"ØØØØ3C3C3C
3CØØØØ"):: CALL HCHAR(5,3,12
8,27)!Ø8Ø
30 CALL HCHAR(19,3,128,27)::
 CALL VCHAR(5,3,128,14):: CA
LL VCHAR(5,30,128,15)!Ø59
40 DISPLAY AT(9,6)SIZE(19):"
W I P E  -  O U T" :: DISPLA
Y AT(15,3)SIZE(24):"PRESS SP
ACE BAR TO BEGIN" !Ø24
50 CALL KEY(5,K,S):: IF K=32
 THEN 70 !189
60 RANDOMIZE :: B=INT(16*RND
)+1 :: F=INT(16*RND)+1 :: CA
LL COLOR(13,F,B):: GOTO 50 !
Ø24
70 CALL CLEAR :: CALL CHARSE
T :: CALL MAGNIFY(3)!Ø43
80 CALL CLEAR :: CALL COLOR(
1,2,2,2,2,2):: CALL SCREEN(2
)!Ø3Ø
90 TRE=10 :: LAP=Ø :: CAR=15
 :: CRASH=Ø :: POINT=Ø :: TI
ME=25Ø :: SPEED=Ø :: FLAG=Ø
!1Ø7
1ØØ CALL CHAR(95,"FFFF")!194
11Ø CALL CHAR(44,"Ø1C3C7C3Ø2
Ø7Ø6Ø6343F37Ø3Ø3Ø1ØBØF8ØD8F8
D84ØEØØØ6181A2AØFØCØBØ91E1")
!143
12Ø CALL CHAR(96,"ØØØ1Ø1Ø1Ø3
Ø7Ø7ØF",97,"ØFØF1F1F3F3F7FFF
",98,"8ØCØEØFØF8F8F8F8",99,"
FCFCFCFEFEFFFFFF",1ØØ,"ØØØØØ
ØØØØØØØØ1Ø3")!2Ø6
13Ø CALL CHAR(1Ø1,"Ø7ØF1F3F7
FFFFFFF",1Ø2,"ØØØØØØØØØØØ8Ø8Ø
CØ",1Ø3,"CØEØEØFØFØFØFCFEFF")!
Ø12
14Ø CALL CHAR(36,"Ø11B1F1BØ3
Ø7Ø6Ø4343F37Ø3Ø3Ø1ØFØF8ØD8F8
D8ØØEØ6Ø2Ø2CFCECØCØ8ØFØFØ")
!222
15Ø CALL CHAR(12Ø,"Ø1131F231
F471FØ73F8F7F1FØ5Ø1Ø1Ø1ØØ9ØE
Ø88FØC4F8C2F8F1FEF8AØ8Ø8Ø8Ø"
)!Ø21
16Ø CALL CHAR(124,"ØØØØØ31CE
Ø9CAØACA5BC839CFØ3Ø8Ø8ØØF718
1714949497 1ØF7181Ø1Ø1ØØØØ")!
Ø65
17Ø CALL CHAR(112,"FFFFFFFFF
FFFFFFF"):: CALL COLOR(11,2,
2)!157
18Ø CALL CHAR(33,"FFFFFFFFFF
FFFFFF",4Ø,"FFFFFFFFFFFFFFFF
")!217
19Ø CALL CHAR(1Ø4,"FFFFFFFFF
FFFFFFF")!Ø62
2ØØ CALL CHAR(128,"Ø1Ø1Ø1Ø1Ø
1Ø1Ø1Ø1Ø1Ø1Ø1Ø1Ø1Ø1Ø18Ø8Ø8
Ø8Ø8Ø8Ø8Ø8Ø8Ø8Ø8Ø8Ø8Ø8Ø8Ø"
)!164
21Ø FOR T=1 TO 24 STEP 2 ::
CALL HCHAR(T,7,33):: CALL HC
HAR(T,15,33):: NEXT T !Ø2Ø
22Ø FOR T=2 TO 24 STEP 2 ::
CALL HCHAR(T,7,1Ø4):: CALL H
CHAR(T,15,1Ø4):: NEXT T !117
23Ø CALL VCHAR(1,16,112,2)::
```

(See Page 26)

Afraid of lions, tigers and bears? Obviously, you didn't see the

DINOSAURS!

DISK OF DINOSAURS is unlike anything you've ever seen before. It is an amazing collection containing pictures of real and cartoon dinosaurs, a complete dinosaur alphabet (or "font") for creating dinosaur-documents, background scenes where you can paste the dinosaur pictures anyway you like, a dinosaur hunting license that entitles you to go after REALLY BIG Game, and even four wonderfully detailed, and amusing dinosaur cartoons. All artwork can be used in programs like *TI-Artist*, *Calendar Maker 99*, *Font Writer II*, and any program that uses TI-Artist format artwork.

Now you can "dinosaur-ize" anything from calendars to greeting cards to reports. Create dinosaur stickers, dinosaur pictures, and even dinosaur logos faster then you can say "Brontosaurus".

We do have to put an age limit on these things though - only dinosaur enthusiasts from 5 to 95 are allowed to buy this (unless you can produce a permission slip from your parents). At only $12.95 PLUS $.75 SHIPPING AND HANDLING for two-disks of dinosaur-heaven it won't take anyone long to save their pennies for it. Heck, a dinosaur steak would probably cost more then that. Now you can get a whole zoo of them (cages not included), courtesy of Ken Gilliland and his electronic paint-brush.

Send to: **Asgard Software**
P.O. Box 10306
Rockville, MD 20850

only $12.95 & $.75 S&H

Requires: TI-99/4A, 32K memory expansion, one disk drive. Printer recommended. Corcomp or TI disk controller required to run cartoons. TI-Artist or program capable of using TI-Artist artwork required to use pictures fully. An active imagination strongly recommended.

Credit Card orders: Call Disk-Only Software, 1-800-456-9272. Visa, MC, AmEx, Choice accepted

WIPE-OUT—

(Continued from Page 24)

```
 CALL VCHAR(1,17,112,2):: CA
LL VCHAR(1,32,112,2):: CALL
VCHAR(1,31,112,2)!038
240 FOR T=1 TO 6 :: CALL VCH
AR(1,T,112,2):: NEXT T !066
250 FOR T=8 TO 14 :: CALL VC
HAR(1,T,40,24):: NEXT T !125
260 DISPLAY AT(1,16):"`bdf`b
df`bdf" :: DISPLAY AT(2,16):
"acegacegaceg" !161
270 CALL VCHAR(1,30,112,2)::
 CALL COLOR(9,2,2)!027
280 CALL DELSPRITE(ALL):: IF
 FLAG>0 THEN GOSUB 920 !231
290 CALL SPRITE(#20,120,2,20
,25,#21,120,2,50,130,#22,124
,2,80,25)!084
300 CALL SPRITE(#1,36,2,165,
80,#2,36,2,40,58)!067
310 CALL SPRITE(#16,128,15,1
67,44,#17,128,15,167,110)!22
8
320 CALL SPRITE(#3,36,2,10,7
7,#4,36,2,90,97)!025
330 DISPLAY AT(4,22):"GEAR"
:: DISPLAY AT(6,20):"1 2 3 4
" !136
340 DISPLAY AT(9,18):"TIME="
;TIME !183
350 DISPLAY AT(10,24):"___"
!169
360 DISPLAY AT(14,18):"SPEED
= ";SPEED :: DISPLAY AT(15,2
6):"___" !188
370 DISPLAY AT(17,16):"CRASH
ES=";CRASH :: DISPLAY AT(18,
25):"__" !216
380 CALL SCREEN(4)!149
390 DISPLAY AT(24,16):"*K* T
O START" !171
400 CALL COLOR(1,11,4,2,2,4,
11,5,4,9,2,5):: CALL SCREEN(
4):: CALL COLOR(#20,13,#21,1
3,#22,2,#1,16,#2,11,#3,4,#4,
16)!135
410 CALL KEY(5,K,S):: IF K<>
75 THEN 410 !218
420 DISPLAY AT(7,22):"_" ::
LM=2 !144
430 SPEED=40 !093
440 CALL SOUND(100,-2,1,110,
1):: DISPLAY AT(24,16):"" !1
67
450 CALL MOTION(#2,CAR,0,#3,
CAR-2,0,#4,CAR+2,0,#20,TRE,0
,#21,TRE+1,0,#22,TRE,0)!071
460 DISPLAY AT(14,25):SPEED
!056
470 CALL MOTION(#1,0,0)!253
480 TIME=TIME-1 :: IF TIME<1
 THEN 660 :: LAP=LAP+LM :: D
ISPLAY AT(9,23):TIME !114
490 CALL PEEK(-31877,A):: IF
 A AND 32 THEN 590 !130
500 CALL KEY(5,K,S):: IF S=0
 THEN 470 !032
510 IF K=75 THEN CALL MOTION
(#1,0,-8):: GOTO 480 !236
520 IF K=76 THEN CALL MOTION
(#1,0,8):: GOTO 480 !043
530 IF K=49 THEN GOSUB 580 :
: DISPLAY AT(7,20):" " :: CA
R=-5-(FLAG*5):: SPEED=10 ::
```

(See Page 28)

PRESENTS THE

6TH ANNUAL

T I Faire Convention

Supporting the TI 99/4A Computer at

HOLIDAY INN

3505 Algonquin Rd.(Rt. 62), Rolling Meadows,IL

Easy Access from I-90 & Rt. 53

Exhibits and Seminars

Saturday, Nov. 12, 1988
9:00 A.M. to 6:00 P.M.
Admission - $4.00

Social Mixer
Friday, Nov.11
8:00 - 12:00 P.M.
Price $4.00

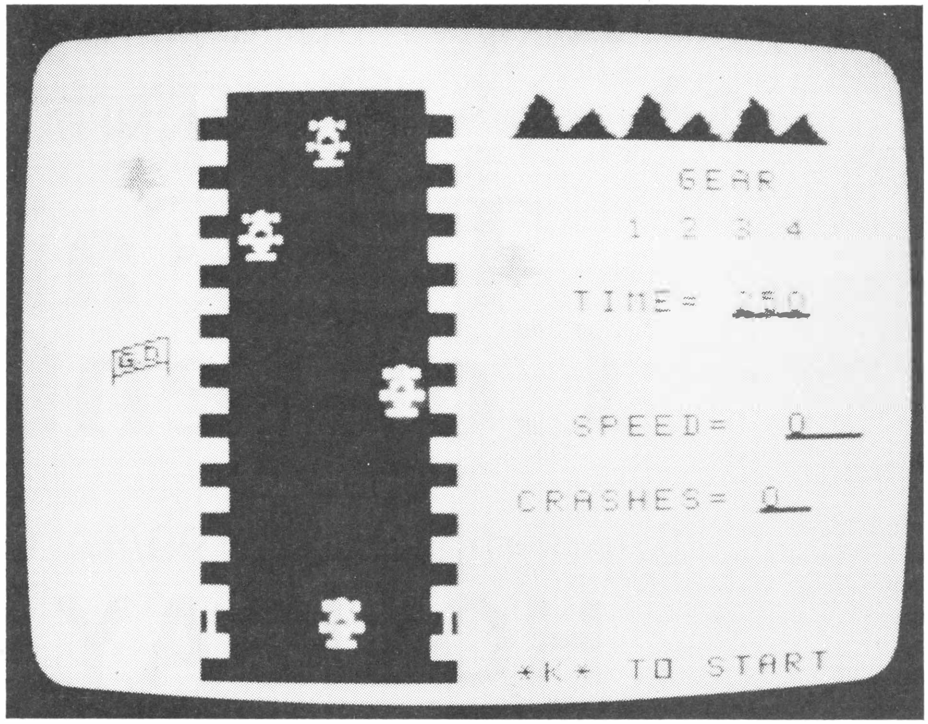
Dinner
Sat. Nov 12
7:00 - 9:30 P.M.
Price $10.00

For more info on the TI FAIRE or for Membership-Write:
Chicago TI Users Group,
P.O.Box 578341, Chicago,IL 60657
or Call our Hotline: 312/755-0051

WIPE-OUT—

(Continued from Page 26)

```
TRE=6 :: LM=1+FLAG :: GOTO 4
50 !110
540 IF K=50 THEN GOSUB 580 :
: DISPLAY AT(7,22):"_" :: CA
R=15+(FLAG*5):: SPEED=40 ::
LM=2+FLAG :: TRE=10 :: GOTO
450 !007
550 IF K=51 THEN GOSUB 580 :
: DISPLAY AT(7,24):"_" :: CA
R=20+(FLAG*5):: SPEED=85 ::
LM=3+FLAG :: TRE=16 :: GOTO
450 !022
560 IF K=52 THEN GOSUB 580 :
: DISPLAY AT(7,26):" " :: CA
R=28+(FLAG*5):: SPEED=110 ::
 LM=4+FLAG :: TRE=22 :: GOTO
 450 !069
570 GOTO 480 !048
580 CALL MOTION(#1,0,0):: DI
SPLAY AT(7,20):"         " ::
CALL PEEK(-31877,A):: IF A A
ND 32 THEN 590 ELSE RETURN !
205
590 CALL COLOR(#1,7):: CALL
MOTION(#1,0,0):: CALL SOUND(
100,-5,1,110,1):: CRASH=CRAS
H+1 :: DISPLAY AT(17,16):"CR
ASHES=";CRASH !253
600 FOR T=2 TO 28 :: CALL MO
TION(#T,0,0):: NEXT T !195
610 CALL PATTERN(#1,44):: CA
LL SOUND(-100,110,1,-7,1)!17
8
620 CALL LOCATE(#2,40,58)::
CALL LOCATE(#3,10,77):: CALL
 LOCATE(#4,90,97)!006
630 FOR T=1 TO 20 :: CALL SO
UND(-50,110+T,T,-5,T):: NEXT
 T :: CALL PATTERN(#1,36)::
CALL COLOR(#1,16)!099
640 CAR=15+(FLAG*5):: TRE=10
 :: SPEED=40 :: DISPLAY AT(7
,22):"_" :: LM=2 :: CALL LOC
ATE(#1,165,80):: GOTO 450 !1
76
650 CALL MOTION(#1,0,0):: CA
LL SOUND(200,700,1,-2,1,110,
1):: CRASH=CRASH+1 :: DISPLA
Y AT(17,16):"CRASHES=";CRASH
 :: CALL LOCATE(#1,165,80)::
 GOTO 450 !215
660 IF CRASH<2 THEN FLAG=FLA
G+1 :: CALL SOUND(50,800,0):
: CALL SOUND(50,700,0):: CAL
L SOUND(60,800,0):: TIME=100
 :: CAR=15+(FLAG*5):: GOTO 2
80 !137
670 FOR T=1 TO 5 :: CALL DEL
SPRITE(#T):: CALL SOUND(-400
,300+T,1):: CALL SOUND(-100,
200+T,1):: NEXT T :: CALL DE
LSPRITE(ALL)!170
680 FOR T=1 TO 14 :: CALL CO
LOR(T,2,2):: NEXT T :: CALL
MAGNIFY(3):: CALL CLEAR !033
690 CALL CHAR(128,"0000000707
0277EFCFC7E277070000000000000
0F0043F37B7F7F7BF343F0F000000"
,104,"FFFFFFFFFFFFFFFF"):: C
ALL COLOR(10,2,2)!038
700 CALL HCHAR(15,1,104,32):
: CALL HCHAR(19,1,104,32)::
FOR T=16 TO 18 :: CALL HCHAR
(T,1,40,32):: NEXT T :: CALL
 SPRITE(#1,128,2,124,200)!22
0
710 DISPLAY AT(13,1):"`bdf`b
df`bdf`bdf`bdf`bdf`bdf" :: D
ISPLAY AT(14,1):"acegacegace
gacegacegacegaceg" !013
720 FOR T=1 TO 2 :: CALL VCH
AR(13,T,112,2):: NEXT T :: C
ALL VCHAR(13,32,112,2):: CAL
L VCHAR(13,31,112,2)!171
730 CALL SPRITE(#2,120,2,150
,100,0,15):: CALL CHAR(119,"
FFFFFFFFFFFFFFFF")!222
740 FOR T=1 TO 12 :: CALL HC
HAR(T,1,120,32):: NEXT T ::
FOR T=1 TO 8 :: CALL COLOR(T
,2,2):: NEXT T :: CALL COLOR
(12,2,2)!182
750 CALL COLOR(11,2,2):: FOR
 T=20 TO 23 :: CALL HCHAR(T,
1,119,32):: NEXT T !049
760 CALL VCHAR(13,1,120,2)::
 CALL VCHAR(13,2,120,2):: CA
LL VCHAR(13,32,120,2):: CALL
 VCHAR(13,31,120,2)!132
770 DISPLAY AT(3,7):"G A M E
     O V E R" !210
780 IF CRASH=0 THEN CRASH=1
!133
790 LET POINT=INT((LAP*3)/CR
ASH)!079
800 IF POINT>HS THEN HS=POIN
T !252
810 DISPLAY AT(6,5):"POINT T
OTAL=";POINT !224
820 DISPLAY AT(9,5):"HIGH SC
ORE=";HS !129
830 DISPLAY AT(24,5):"PLAY A
GAIN Y\N?" !131
840 CALL CHAR(89,"0044442810
101010")!215
850 FOR T=3 TO 8 :: CALL COL
OR(T,16,5):: NEXT T :: CALL
COLOR(10,16,5,12,5,5,11,4,5,
```



(See Page 29)

Texas Instruments Video Chips

The 9918A and earlier chips

By TONY LEWIS

The following is the start of a multi-part series on the TMS 9918A and Yamaha 9938 video chips. The 9918A is used in the 99/4A and the 9938 is used in the Myarc Geneve. — Ed.

Texas Instruments is an industry leader in developing video display processors, or VDPs, as they are commonly called. TI has released three generations of video chips since the mid-1970s: the 9918 series, the 9938 series (fully developed and sold by Yamaha), and the state-of-the-art 340XX series. This and future articles will discuss the TI video chips, their history and implementation in various computers. The main focus will be on the hardware aspects of the VDPs, rather than the software used to run them. Hopefully, readers will gain insight on not only how the video chips work in their computers, but also some of the ins and outs of why programs written to access the VDP use certain addresses and other tricks.

Before going further, please note that I am not an electrical engineer, so if you run across a mistake or two, don't hesitate to bring it to my attention, and I'll note any corrections in a future article. The reader is urged to consult the extremely well-written data manuals on these fine chips, and the TI99/4A Console Technical Data manual, both available directly from Texas Instruments.

THE 9918A ARCHITECTURE

The TMS9918A/9928A/9929A video display processors are display devices that were specifically designed to be used with a color monitor or television set. Previous video chips were made for use with monochrome monitors for pure display of text. The 9918 is indeed a processor itself, which generates all the necessary video, control and synchronization signals for the monitor with little or no extra circuitry. It also controls the dynamic RAM which holds the display information. All of this without any input from the main microprocessor. The 9928A and 9929A are functionally identical to the 9918A, except that the '28 and '29 are for use with RGB (red/green/blue) monitors; the '28 is for American monitor standards, while the '29 was designed for European standards. The bulk of this article centers around the 9918A, which is used in the 99/4A.

The 9918A offers the following features:

- 256x192 resolution
- 15 colors plus transparent
- general 8 bit data bus interface to the main microprocessor
- direct wiring to 4K, 8K or 16K dynamic RAM memories
- automatic and transparent refresh of dynamic RAMs
- multiple VDP systems capability
- external VDP input capability
- composite video output (9918 only)
- unique "plane" representation for 3D simulation (sprites)

Most of these features should alrady be known by most TI users. The 99/4A's VDP can indeed display pixels in a 256 column by 192 row screen with 15 colors plus a 16th 'color,' which is transparent, and equal to the background color. Sprites are also a well knoiwn feature, and are used in most video games. They allow a kind of 3 dimensional image in that sprites appear to pass over and under other sprites, depending upon their assigned number.

The hardware features of the chip are not as well known by most users. The 9918A is always used as a 'memory-mapped' device, which means that the VDP itself is wired to appear as a series of memory locations to the computer (more on that later). The VDP does not depend upon the main processor to control the dynamic RAM or display processes.

As noted above, the 9918A can use 4K or 8K RAMs, but usually 16K x 1 DRAMs are used, as in the 4A. These video chips have some unusual features not often used, such as the ability to intermix the output of two VDPs on the same screen. They can also be wired to accept external video and mix the 9918A output over the video, which becomes the background. There was an article in *Radio Electronics* a couple of years back that described a kit based on the 9918A that allowed the user to superimpose titles and graphics over videotape. These features would be extremely difficult to implement using the 99/4A, unfortunately.

INTERFACING THE 9918A

As mentioned, the 9918A is used as a memory mapped device (Fig. 1). This means that the 9900 can read or write to the VDP as if it were just another RAM memory location; the 9900 does not know that the 9918 is a separate processor, or exactly what it is doing at any given moment. This feature has its advantages and well as disadvantages. The 9900 communicates with the VDP by use of 8 data lines, 3 control lines and an interrupt line. Through this interface the micro can conduct four operations:

1. Write data bytes to Video RAM (VRAM)
2. Read data bytes from VRAM
3. Write to one of eight VDP write-only registers
4. Read the VDP status register

Each of these operations requires one or more data transfers to take place over the data bus, with the three control lines determining the type of operation.

The type and direction of data transfers are controlled by the CSW, CSR and MODE signals. CSW is the 9900 to VDP write select, while CSR is the VDP read select. When either is low (voltage=0

(See Page 30)

WIPE-OUT—

(Continued from Page 28)

```
9,2,5)!167
860 CALL COLOR(1,16,5,9,2,5)
!249
870 CALL COLOR(#1,16,#2,13)!
189
880 CALL KEY(5,K,S):: IF S=0
 THEN 880 !188
890 IF K=89 OR K=121 THEN CA
LL CLEAR :: CALL DELSPRITE(A
LL):: GOTO 70 !119
900 IF K=78 OR K=110 THEN CA
LL CLEAR :: END !116
910 CALL SOUND(100,110,1,150
,1,170,1):: GOTO 880 !065
920 DISPLAY AT(20,17):"EXTEN
DED" :: DISPLAY AT(21,19):"G
AME" :: CALL SOUND(100,700,1
):: CALL SOUND(100,500,1)::
CALL SOUND(200,800,1):: RETU
RN !030
```

Texas Instruments Video Chips

(Continued from Page 29)

volts), the VDP can be read or written to. The MODE pin is usually high (5 volts); is brought low when reading or writing data to the VRAM. A neat trick is to tie the MODE pin to a low order address pin, such as A14, like on the 99/4A. Then the difference between a read/write to the status registers vs. the VRAM is only hex >0002. That way, the addresses used to access the chip automatically take care of the MODE pin status, instead of the software.

You can either read or write to the VRAM to change what is on the screen or stored in memory. But the VDP does not have address lines like ordinary RAM, so how can you tell it to write or read a byte out of 16K worth of VRAM? This is done by sending a 14 bit address first, one byte at a time, then reading or writing the data. For example, a video write would take 3 byte transfers over the 8 bit data bus to complete:

Byte	Data transferred
1	address bits A6-A13
2	address bits A0-A5
3	data bits D0-D7

So every time the 9900 wants to read or write to just one byte in the video memory, it has to transmit both the address and data as data bits to the 9918A, which interprets the information.

Reading and writing to the VDP registers are just a little less complicated. To write to one of the eight registers, you send the 8 bit data as the first byte, then the second byte contains the 3 bit register-select code. To read the VDP status register, you merely read the 8 bit byte. Remember that while you're reading and writing to registers and VRAM, the 3 control signals must also be to the appropriate values (high or low) to determine exactly what's going on.

A few other interface pins exist on the 9918A to keep things coordinated between the micro and the VDP. The INT interrupt pin is used to generate a low interrupt at the end of each active- display scan which is about 1/60 second for the 9918A. This interrupt can be turned on and off by control and status bits in various registers. The RESET pin caused the VDP to reset itself whenever the reset signal is brought low. This allows the VDP to get ready when the computer is first turned on, or when manually reset by a user-activated switch.

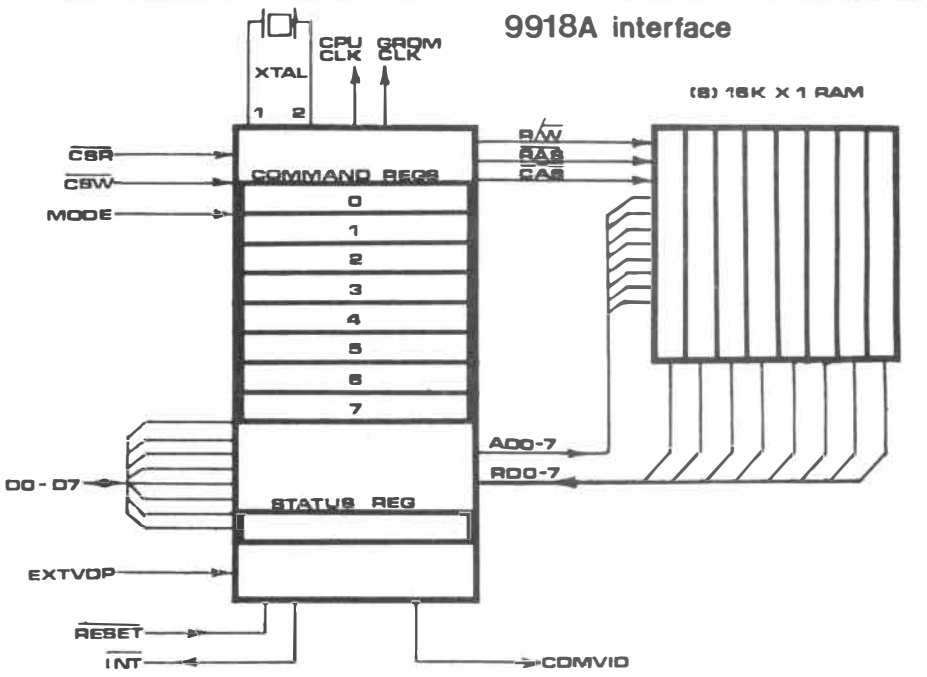


Fig. 1

Timing for the chip comes from the XTAL1 and 2 pins, where a 10.7+ Mhz crystal is connected. The CPUCLK pin is the color burst pin, and outputs a signal at the crystal frequency divided by 3. The GROMCLK signal is output at a frequency equal to the crystal frequency divided by 24. We'll show you how they are used on the 4A later.

OTHER VDP INTERFACES

The 9918A interfaces to the VRAM (which is just ordinary dynamic RAM) by means of two sets of complex function pins: RD and AD. When we say that the VDP can address 16K of memory, we are talking in terms of bytes, or 8 bits. But, unlike the commonly available static RAM which often comes in 8K x 8 or 32K x 8 formats, dynamic RAM usually comes in a 1 bit format. Therefore, to get a byte, or 8 bits of data, we need eight 16K x 1 VRAMs. So the 9918A is connected to 8 chips to get a byte of information each time.

Another unusual aspect of dynamic RAM is its addressing modes. The VDP has only 8 address bits, AD0-AD7. How can 8 address bits reach to 16K worth of addresses? The answers is provided by the two pins marked RAS and CAS. Dynamic RAM addresses are split into two components — rows and columns. Unlike a static RAM (which inputs all address lines simultaneously), a dynamic RAM will receive the first part of the address on AD0-AD7 when RAS goes low, then the second part on the same AD0-AD7 lines when the CAS line goes low.

Needless to say, it all happens quite quickly. Be grateful that you do not have to program the microprocessor to do this!

Since dynamic RAM needs to be refreshed on a frequent basis, the VDP also takes care of this bit of housekeeping automatically too (transparent refresh). As you may have suspected, the RD0-RD7 lines are the data lines that are connected to the individual data pin (one per chip, not 8) of each of the 8 VRAM. The RD lines read data output from the RAM; lines AD0-AD7 are used to write to the RAM, in addition to being address lines. A R/W signal tells the VRAM if the VDP wants to read or write a byte to the VRAM.

The COMVID pin outputs the composite video for use with a composite monitor (Fig. 2). A simple external modulator circuit revises this signal to allow its use with a color TV set. The EXTVDP pin allows input from another VDP, such that their signals can be intermixed. The RESET pin mentioned earlier is also used in synchronizing external video signals, like those of a VCR, with the VDP output. In this case, the external video becomes the background, and the text or sprites/grahics

(See Page 32)

Texas Instruments Video Chips

(Continued from Page 30)

is superimposed over the external video.

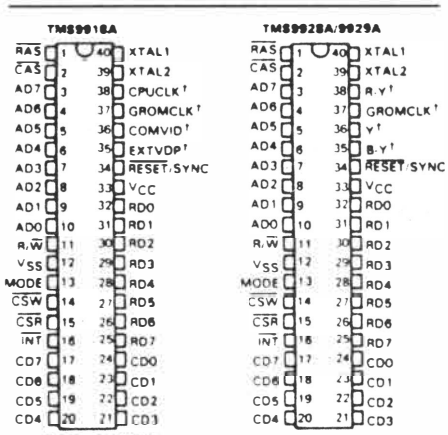


Fig. 2: 9918A/28A pinouts

VDP Registers

Unlike other general purpose microprocessors, the video display processor can not be manipulated by running its own program. With the 9918A, the programmer can cause the various features of the chip to be enabled or disabled by setting the individual bits, or "flags" within registers. Likewise, the VDP can tell you what's going on by setting flags within its status register, which you can read.

The eight VDP write-only registers are defined by the user to give the desired results. There are numerous articles and books on what each bit of each register is for, so we won't belabor the definitions here. Instead, here's a short summary of the eight registers and some of their features:

Register 0: two bits (out of eight) are used to control the MODE and external VDP input.

Register 1: has eight option bits which control various features such as interrupt enable, BLANK enable, and the operating mode of the chip. Sprite size and magnification are controlled here too.

Register 2: defines the base address of the Name Table sub- block.

Register 3: defines the base address of the Color Table sub- block.

Register 4: defines the base address of the Pattern, Text or Multicolor sub-block.

Register 5: defines the base address of the Sprite Attribute Table.

Register 6: defines the base address of the Sprite Pattern Generator sub-block.

Register 7: contains the color codes for the text and background in the Text mode.

The Status Register is an eight bit register that is read by the 9900 to determine the status of various items. You cannot write to this register. The Status register holds the following information:

- Interrupt flag: this is set to 1 with each interrupt, and cleared to 0 when the register is read.
- Coincidence flag: this is set to 1 when two or more sprites coincide (have an overlapping pixel); it is also cleared when the Status register is read. Good for game applications.
- Fifth sprite flag and number: if there are five sprites on a horizontal line, the flag is set to 1, and the number of the fifth sprite is placed in the lower 5 bits of the register. Once again, good for game applications.

VIDEO DISPLAY MODES

The video display modes available with the 9918A are all controlled by software, and are mentioned briefly here for com-

(See Page 34)

1988 The Year Of The Upgrade Not The Up Trade

A COMPLETE EXPANSION SYSTEM 9900 + FOR THE TI99/4A THE ULTIMATE 99/4A EXPANSION SYSTEM AT A SPECIAL PRICE FROM TEX-COMP™

CorComp

COMPLETE EXPANSION SYSTEM NOTHING ELSE TO BUY!

TEX-COMP, the undisputed leader in supplying the 99/4A User, has now put together the finest and most complete expansion system ever offered for the TI99/4A.

- **9900 Expansion Box & Regulated Power Supply (UL Approved)**
- **32K Memory Upgrade** Adds 32K bytes of Random Access Memory to your system.
- **Double Sided/Double Density Disk Controller** (operates up to 4 drives)
- **RS232 Interface** Lets you add a wide range of other accessories, such as printers or telephone modems, one parallel and 2 serial outputs.
- **1 SS/SD Disk Drive** Allows you to store and retrieve data on 5¼-inch single- or double-sided flippy diskettes.
- **1 Disk Drive Case & Regulated Power Supply** Handles two ½-height drives easily (UL or LAC Approved)
- **New Disk Manager with Improved Disk Utilities**
- **All Cables & Instructions** Including a free TI RS232 Y-Cable.

ALL FOR $379.95 Plus S&H

For above system with full size or ½-height DS/DD Drive	**SPECIFY**	$399.95
For above system with a pair of ½-height DS/DD Drives	**NEW LOW PRICE**	499.95

CorComp RS232 Card (for TI P-Box)		**79.95**
CorComp 32K Card (for TI P-Box)		99.95
CorComp 256K P-Box Card		WRITE
CorComp 512K P-Box Card		WRITE
CorComp DS/DD Controller (for TI P-Box)		**149.95**
CorComp 9900 System with Free RS232 Y-Cable		**299.95**
CorComp Stand Alone RS232 with Free Y-Cable		**99.95**
CorComp Stand Alone 32K		**89.95**
Triple Tech P-Box Card (Clock/Buffer)		119.95
9900 Clock Stand Alone		**69.95**
"Grom Buster" (for 1983 Consoles)		24.95
Load Interupt Switch (with **FREE** Screen Dump Program)		19.95
PDI Diagnostic Module		**24.95**
NEW "Writer-Ease" Word Processor & Spell Check	New Dictionary	**39.95**
IBM Connection (39.95 with Controller Card)		**49.95**
½-Height DS/DD Disk Drive (2 will fit in P-Box)	**New Lower Price**	**89.95**
Full Size SS/SD Disk Drive (exact replacement for TI 1250)		**59.95**
Full Size DS/DD Disk Drive		**79.95**
Drive Enclosure with Regulated Power Supply for 2 ½-height or 1 full Drive		**39.95**
Cable Kit for 2 ½-height Drives (for installation in P-Box) (9.95 with drives)		**29.95**

Send order and make checks payable to

TEX-COMP

P.O. BOX 33064 – GRANADA HILLS, CA 91344

MasterCard VISA

VISA and MASTERCARD HOLDERS CALL DIRECT

(818) 366-6631

24 Hour Order Line

TERMS: All prices F.O.B. Los Angeles. For fastest service use cashiers check or money order. Add 3% shipping and handling ($3.00 Minimum). East of Mississippi 4½%. Add 3% for Credit Card orders. Prices and availability subject to change without notice. We reserve the right to limit quantities.

NOTE: Payment in full must accompany all orders—credit card, company check or money order for immediate shipment. Personal checks require up to 4 weeks to clear. California orders add 6½% sales tax.

Texas Instruments Video Chips

(Continued from Page 32)

parison with the later VDP chips. An interesting feature of the 9918A was (for its time) not only the built-in ability to display color, but the ability to display text **and** graphics, even a bit-map mode. Sprites, of course, are another built-in feature that made games and advanced graphics a snap, since the VDP controlled manipulation of the individual sprites. If you have a LOAD switch for your console, you may have seen the sprites still moving, even though the rest of the screen was frozen by pressing the switch. That is because the VDP keeps track of the sprites, not the 9900.

The display modes are Graphics I, Graphics II, Multicolor, and Text. Graphics I and II allow the familiar 32-column text display with graphics, with II mode capable of a more complex display. Multicolor mode simply displays individual blocks of color, while Text mode allows a 40-column display, often used in assembly- based programs.

The 9918A AND THE 99/4A

Well, now that we have covered the ins and outs of the 9918A, let's look at how it is used in the 99/4A (Fig. 3).

The registers and the display modes are, of course, the same for the 9918A in the 4A. For the 8 bit data bus, the TI uses only the D0-D7 lines. Address line 14 is connected to the MODE pin; that is why the read and write addresses are separated by >0002 (ie. VDP write data= >8C00, VDP write address= >8C02).

The various register/addressses are set by a 74LS128 3 to 8 line decoder, which is a terrible waste of address space (>8000- >9FFF), but then the TI was designed over ten years ago, before modern programmable logic devices were in common use. The VDP interrupt signal is routed to the 9901, and is input as an -INT2, or second level interrupt. The VDP reset pin is tied to the system reset pin to keep things synchronized. The GROMCLK signal goes — where else — to the GROMs for timing purposes. The CPUCLK signal goes to the TMS9919 sound generator for a clock signal. The VRAM used are eight TMS4116 16K x 1 dynamic RAMs, configured as previously noted. As a testament to their relative age, the 4116s need not only 5V for power, but -5V and 12V as well.

One thing more to note is the use of VRAM for storage of BASIC programs, PABs, buffers, etc. The TI console itself does not have much RAM to call its own, just a small block at >8300->83FF. TI shoved everything else into the VRAM. Therefore, in addition to being so slow in executing BASIC programs because of double interpretation, the poor 9900 has to slow down to talk with the 9918A, since it is memory-mapped. Even with assembly programs, the relatively useless opcode

(See Page 35)

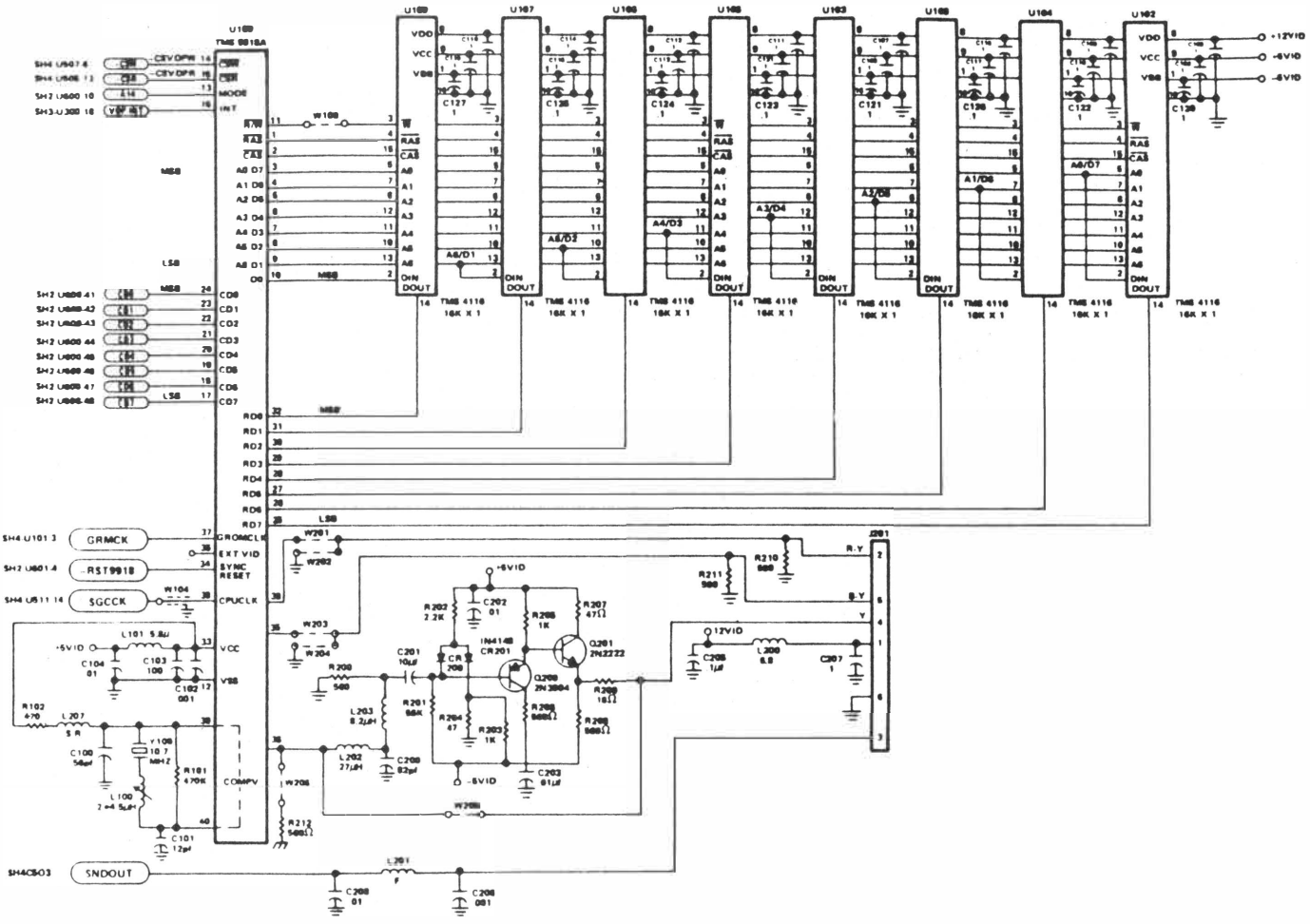


Fig. 3 The 99/4A Video Subsystem

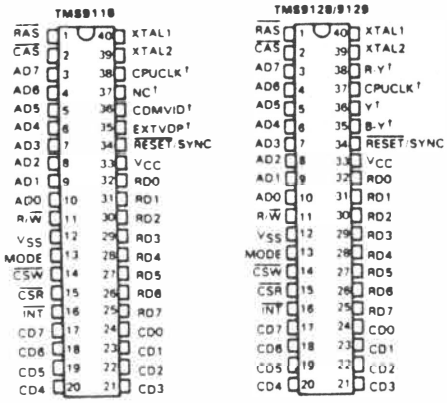
Texas Instruments Video Chips

(Continued from Page 34)

"NOP" (no operation) is often used in VDP accesses to allow for timing differences between the 9900 and 9918A.

OTHER TI VIDEO CHIPS

Technically speaking (pun intended), the 9918/28/29A series does not exist anymore. They were replaced in the early 1980s with the 9118/28/29 series. The 9118 is a drop-in replacement for the 9918 *except* that the GROMCLK isn't available (Fig. 4). The other major change is that the 9118 can use 16K x 4K DRAM, instead of 16K x 1K, thereby cutting the chip count down by six (Fig. 5).



Pins 35 to 38 are the only pins which vary for each device

Fig. 4: 9918/28 pinouts

But before the 9118 and the 9918A, there was a TI video chip called the TMS9927/37 (Fig. 6). Not many people have heard of it, but it was a powerful chip, more powerful than the 9918A in some ways. The 9927 was a pure monochromatic text generator that could be programmed to display up to 132 columns per line, and up to 64 lines per frame, if the monitor could accept them! Figure 7 shows the typical layout.

So, why don't we use that? Because the scrolling and cursor controls are primitive, and useful more for simple data displays, rather than in a personal computer. And the 9927 has a messy interface, requiring dual-port RAM, a character generator ROM, etc. A product of the 1970s, the 9927 was state-of-the-art ten years ago, but not today.

NEXT?

That's a quick rundown on the TMS9918A from a hardware viewpoint. Next we'll cover the 9938, the second

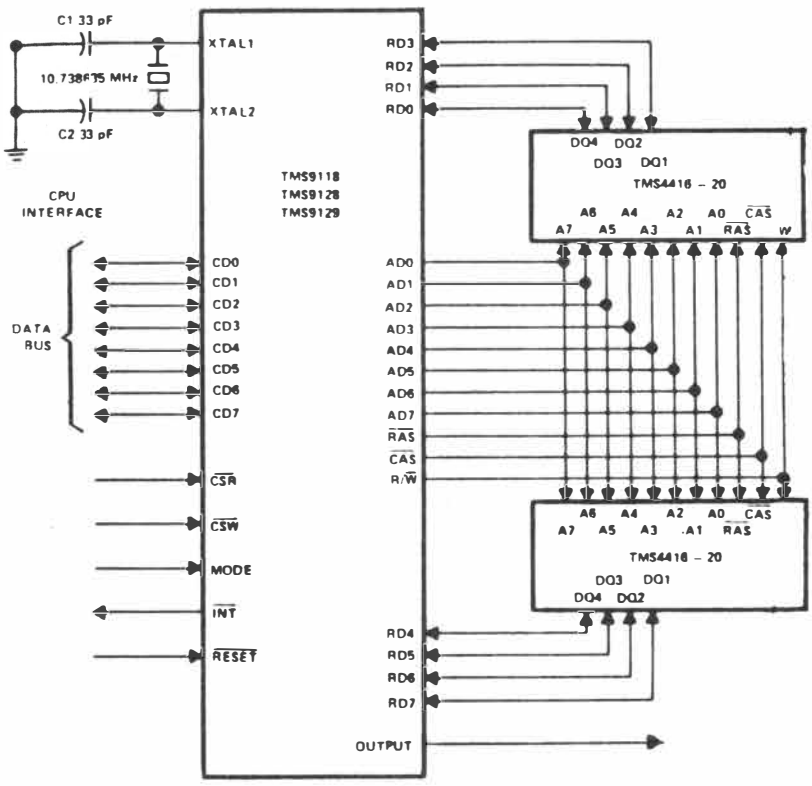


Fig. 5: 9918 to VRAM interface

generation VDP that was initiated by TI, but completed by Yamaha and Microsoft, and available today.

If you'd like to know about the 9918A (and other) chips, consult the references below. Questions about this article may be sent directly to the author at 409 Drolmond, Raleigh, NC 27615 or on Compuserve at CIS 73357,1730.

REFERENCES

TMS9918A/28A/29A Video Display Processors Data Manual, 1982

TMS9118/28/29 Video Display Processors Data

(See Page 36)

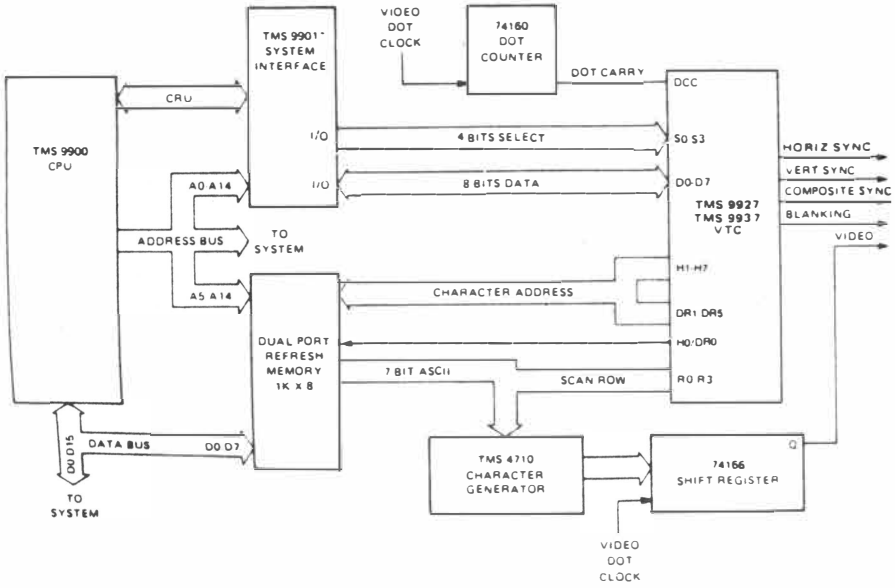


Fig. 7: 9927-based video system

Geneve

Myarc's Alabama connection

By MIKE DODD

Many people have written or called me complaining about Myarc customer service. The complaints all have one thing in common: people have been having an extremely difficult time getting answers from Myarc.

Almost all Myarc literature lists the New Jersey address and phone number for Myarc, but, according to many users, getting through to the New Jersey office on the phone has been extremely difficult at times — I understand that the New Jersey office does not have a full-time secretary, and is answered whenever New Jersey Myarc employees have time. Some users have said that the phone has been busy for very extended periods of time, leading them to suspect that the phone was left off the hook.

According to Myarc spokesman Jack Riley, customer service is now being handled by the Alabama office. Any written communication sent to New Jersey is forwarded once a week to the Alabama office, Riley says. You can call or write the Alabama office directly, in an effort to expedite the response. The address is: Myarc, Inc., 2624 Ranier Drive NE, Birmingham, AL 35125. The phone number is 205-854-5843. The Alabama office phone will be answered by a secretary or Riley.

According to Myarc, return authorizations must still be obtained from the NJ office.

Some users have said that even when they do get through to customer support, the answers given have been unsatisfactory. One user said that he was repeatedly told that his problems would be fixed with new system software, when, in fact, the keyboard turned out to be at fault.

One common complaint is that many users have trouble understanding the literature Myarc supplies. As general purpose advice, you should carefully read and re-read the documentation supplied. The answers are often in there, but they may be difficult to find. I have gotten several letters asking about some feature that turned out to be covered in the documentation.

It seems that many of Myarc's customer support problems have to do with a lack of adequate resources to handle the large volume of inquiries — Myarc is a small company. Riley told me that MICROpendium will soon start a monthly column in which Myarc will answer general purpose questions sent to MICROpendium.

Users are also invited to call me (615-435-4169) or contact me on DELPHI. (If you don't want to call me, you can write me; however, a written response cannot be guaranteed. A written response is much more likely, however, if the sender encloses a SASE.) To contact me on DELPHI, where I have recently been set up (thanks to system manager Jeff Guide), my ID is MY9640. Paul Charlton (ID TI994A) and J. Peter Hoddie (ID GENEVE) have also recently been set up on DELPHI (again, thanks to Jeff Guide) — those two can also help with many Geneve-related questions. Users who send me mail on DELPHI or call me will get a much quicker response than those who write. I can often diagnose problems more readily on the phone than I can through mail.

XOPS

In MDOS V1.01 and earlier, some of the XOPs documented in Paul Charlton's XOP manual are not implemented. In addition, in some of the cases where the documentation says that the EQ bit will be set or reset depending on an error condition, the EQ bit is in fact not changed. Two of the XOPs not functional in MDOS V1.01 are sound and SetTTYPos. Several readers have asked about those two calls. These problems should be fixed in later versions of MDOS.

QUESTIONS & ANSWERS

Glenn Davis asks: *Can MDOS load program files larger than 8K from the main prompt?*

MDOS is capable of loading a file that is >3F80 bytes long. As there is a six-byte header at the start of each file, you can load a program segment that is >3F7A bytes long.

Edward Wirick asks *if the CorComp 'Real' TI/IBM Connection program will work on his Geneve.*

The answer is no, for the same reasons that fast disk copy programs will not work (see July 1988 MICROpendium) — the CorComp TI/IBM program uses low level disk access to read the IBM disk and, as written by CorComp, will not work on the Geneve.

PRBASE V2.1 UPDATE

Many people have been having problems with the PRBASE V2.1 update that I wrote. The problems center around the CREATE portion of the program. One major problem that a few people have noticed is that the initialize disk function of CREATE will not correctly format a disk — it does not initialize sector 0 to contain the proper values, and as such, will not work.

Another problem is that CREATE does not read sector 0 of the data disk, which can cause a "Record Not Available" error message on some functions of CREATE (e.g. Tabular Reports). The trick is to get the computer to read sector 0 of the data disk, then load CREATE, without exiting back to the title screen in between those two steps. Two easy methods of accomplishing this are as follows:

- For both methods, place the program

(See Page 37)

TI VDP CHIPS

(Continued from Page 35)

Manual, 1984
TMS9927 and TMS9937 Single-Chip Video Timers/Controllers, 1982
TI-99/4A Console Technical Data, 1983

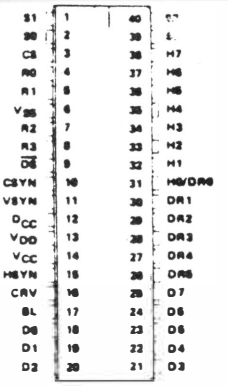


Fig. 6: 9927 pinout

GENEVE—

(Continued from Page 36)

disk in DSK1, and the data disk in DSK2.

• If you are using Extended BASIC to load, go to the XB prompt. Type CALL DIR(2) <ENTER>. This will attempt to catalog the disk, and should report an empty disk. However, in cataloging the disk, it will read sector 0. Then type RUN "DSK1.LOAD" <ENTER> and select CREATE (option 1) from the menu screen.

• If you are using the Editor/Assembler to load, go to E/A option 5. Type DSK2.X <ENTER>. This will generate an I/O error, but will read sector 0. Go back to option 5, then type DSK1.CRT:1, Enter to load CREATE, or just type <ENTER> to load the menu program (DSK1.UTIL1).

If you have only one disk drive, you should insure that your program disk and data disk are the same format. You should then be able to load the CREATE program normally, then switch disks to the data disk. You should not have any problems with doing this.

You do not have to worry about any of this when using the main PRBASE (Data Management) portion of the program — it will read sector 0 automatically on selecting a disk disk to use.

On a related note, when I wrote the changes for V2.1 of PRBASE, I did not fix any of the bugs or quirks that PRBASE V2.0 had. The only change I made was to change the sector usage.

MDOS & DATES

Many users have complained that MDOS does not correctly set the day of week. One solution is to write an MDOS program that, when run, will add one to the day of week. You could then run this program every time you execute the DATE command.

Enter the program with MY-Word, using the (P)rogram (E)dit option. Save the file.

Assemble the saved source file with options 'RC'.

Exit GPL and go back to MDOS. From the command line, run Paul Charlton's fairware Linker program. Linker is available from many BBSes, networks and user group libraries. Please remember that if you use Linker, you should pay Charlton for his efforts.

From the Linker prompt, type the name of the object filename and <ENTER>. Remember that if you used a slash (/) in the file name, you will need to enclose the slash with quotes, so that MDOS does not consider it an option string and thus ignore it.

Now type @filename <ENTER>, where filename is the name that the program should be saved as. For example: @B:FIX will save the program to the B drive (normally DSK2.) with a filename of 'FIX'.

Type ! <ENTER> to exit back to MDOS.

To try the program, type DATE <ENTER> to see what the current day of week setting is. Now type the filename of the saved file (e.g. B:FIX). Again type DATE and observe the change.

For those who are unable to obtain Paul Charlton's Linker program from user groups or electronic bulletin boards, the Boston Computer Society has the program on disk No. 91 in its public domain library. Send $3, plus $1 handling, to: Boston Computer Society, TI99/4A User Group, One Center Plaza, Boston, MA 02108.

Readers with questions about the Geneve or problems may write Dodd at 116 Richards Dr., Oliver Springs, TN 37840. Questions may be answered in this column. Personal replies cannot be guaranteed.

```
* Fix day of week byte
* Copyright 1988 by Mike Dodd
* Written for MICROpendium magazine
*
       LWPI >F000              Load workspace
       MOVB @>F13E,R0          Get day of week byte
* Day of week now in Most Significant Byte of R0.
* There may be garbage bits in the first four bits
* of R0, so we need to clear those.
       ANDI R0,>0F00           Strip all but relevant bits
* Now add 1 to the value. If doing this results
* in a 8, then we need to reset to 1.
       AB   @H01,R0            Subtract one
       CB   R0,@H07            Less than or equal to 7?
       JLE  N0T8               Yes - no problem
       MOVB @H01,R0            Wrap back to 1
* All fixed. Reset the value and return to MDOS.
N0T8   MOVB R0,@>F13E          Reset day of week byte
       BLWP @>0000             Return to MDOS
*
H01    BYTE >01
H07    BYTE >07
*
       END
```

Software schedule announced

Myarc releases 512K Geneve card

Owners of the Myarc Geneve computer can now add 512K of RAM with Myarc's new 512K memory card. Coupled with the Geneve's 640K of memory, users will have more than one megabyte of memory. The card is priced in the $300 range.

Owners who ship their existing Myarc 512K cards configured for use with the TI99/4A can have the cards modified for use with the Geneve, according to Myarc spokesman Jack Riley. However, after the modification, the card will no longer be compatible with the TI. The cost of the modification is $50.

Myarc also recently released its long-awaited Hard and Floppy Disk Controller card for use with the TI and the Geneve. Those who use it with the Geneve, however, are required to use MDOS Ver. 1.06 and GPL Ver. 1.01. Myarc Disk Manager 5 Ver. 1.21 is required for use with either computer.

Riley says that several programs will be released for the Geneve in coming months, including Advanced BASIC, Pascal, GEME, a utility package and MY-Pro-Art, a completely modified version of MY-Art.

Advanced BASIC and Pascal are scheduled for release in September.

GEME is expected to be released in October with a price of about $175. GEME is a multi-tasking, windowing environment that allows users to run up to four programs simultaneously. GEME will support 512 x 424 resolution with the Myarc 512K card. Without the additional memory, the display will be 512 x 212.

The utility package, priced at about $100, will offer users an array of functions. Scheduled for release by October, the package will include several versions of GPL (one will load software without stopping at a loader screen, another will load a user-configured menu); a GRAM Kracker-like utility that will allow users to manipulate software in RAMdisk; and a sector editor for hard and floppy drives that will allow users to recover corrupted or deleted files and some programs.

The utility package will be the first program released with Myarc's utility/protection card containing a read once EPROM. As announced last spring, Myarc will protect all of its software releases and won't release anything prices at less than $100. "This leaves the field clear" for third-party vendors, Riley says. Myarc will burn EPROMs for third-party vendors who wish to incorporate Myarc's protection scheme into software. Each utility card has the capability of reading protection EPROMs for numerous programs.

The protection/utility card also allows the user to plug in a TI speech synthesizer board. It also provides support for an as yet unspecified math coprocessor chip. A MIDI port is also possible, according to Riley. While the card will be bundled with the utility package, it may become available to those who want it for its support of the speech synthesizer.

MY-Pro-Art is "completely different from MY-Art Version 1.4," according to Riley. Like GEME, it will support 512 x 424 screen resolution with a 512K memory expansion. MY-Pro-Art reads and writes in GIF and RLE formats and can import files created by TI-Artist and GRAPHX. It also supports multiple fonts, clipboard functions, color printing, and features a grid function for perspective. Riley said that the final version isn't finished and that Myarc would be receptive to suggestions about what features users would like the finished product to have.

MY-Pro-Art, which requires a mouse, will be available with or without the Myarc mouse. Scheduled for release in October or November, it will be protected

Print-Screen enabled

Improvements made to MDOS, GPL

MDOS Ver. 1.06 has recently been completed and it features a number of improvements over the most widely released MDOS, Version 1.01.

Most noticeable to users will be the fact that it automatically boots in 80-column mode instead of 40 columns. The Print Screen function also works. When a user presses CTRL Print Screen data that cross the screen will be outputted to a printer until CTRL Print Screen is toggled to turn it off.

Beyond this, in conjunction with Version 1.02 of the GPL interpreter, users can use AUTOEXEC files to load GPL as well as program files. For example, the following AUTOEXEC file will load GPL and the Editor/Assembler automatically:

TIMODE
GPL A:EA

Using this AUTOEXEC file, the Editor/Assembler is loaded from drive A. It may also be written as GPL DSK1.EA.

After loading GPL, the user can return to the MDOS A> prompt by pressing CTRL-ALT-DELETE at the GPL selection screen. In previous versions, this command would reboot the system.

Version 1.06 of MDOS provides limited hard disk support. Full hard disk support will be available with Ver. 1.10H. Ver. 1.10H will be identical to Ver. 1.10, with the addition of hard disk support.

The only modification to commands in Ver. 1.06 is with the attribute command. In Ver. 1.06 it is implemented like this: ATTRIB (- or +)P. In earlier versions of MDOS, the command operated in this way: ATTRIB (- or +)R.

GPL Ver. 1.02 differs slightly from Ver. 1.01. According to Myarc, the only difference between these two versions is that Ver. 1.02 boots up with the sound chip turned off while Ver. 1.01 boots with the sound chip on.

MacFlix: Use Macintosh Graphics on your TI

Finally - access to Macintosh™ MacPaint™ graphics using your TI-99/4A or MYARC 9640 Computer! With MacFlix by J. Peter Hoddie you can view, print, and save industry standard MacPaint graphics with your computer. The Macintosh has the most extensive library of graphics ever created for a personal computer, and now you can utilize this vast resource.

Using MacFlix you can load and view MacPaint images. These images are a full 8 by 10 inches so MacFlix gives you the power to easily move around the picture. If you have an Epson compatible or ProWriter printer the image may be printed.

But graphics are useless unless you can use them in your own designs. With a keystroke MacFlix saves your image as a TI-Artist screen which can be manipulated with most TI graphics programs!

MacPaint pictures are available on all major telecommunications networks, many local BBS's, and through user groups.

If you have a 9640 computer, MacFlix is even more powerful. You can view the picture in high-res mode, with or without interlace. On a 9640 MacFlix, can actually display more than a standard Mac screen! MacFlix also saves images in both MY-Art formats.

MacFlix has the extras you've come to expect from Genial Computerware - the ability to invert a picture, and convenient Catalog and Delete File routines. If you own our PC-Transfer program, MacFlix comes with a custom conversion routine that lets you move MacPaint pictures to and from IBM disks, a great source of thousands of pictures.

Written in assembly language, MacFlix runs on a TI-99/4A computer with Extended BASIC or Editor Assembler, or Super-Cart; or a MYARC 9640 computer.

MacFlix is available for only $15. Place your order before September 30, 1988 and receive a free bonus disk of Macintosh pictures to start your collection!

Graphics Expander

Created by J. Peter Hoddie, Graphics Expander is the most versatile graphics utility ever available for users of TI-Artist, CSGD, Font Writer II, and other popular graphics software. Here's a list of the main benefits:

- Handles TI-Artist Fonts and Instances, and CSGD Fonts and Small Graphics.
- Can be used to convert between TI-Artist and CSGD formats
- Can stretch and shrink graphics by user selectable factors from -9 to 9
- Graphic display of all conversions to eliminate guess work
- Provides Upside-down, Invert, Mirror and Rotate features
- Built in Catalog and Delete file options
- Written in 100% assembly language for blazing speed
- Complete printed documentation by Walt Howe

Because we know that TI-Artist can't handle the really big fonts you want to use, we include a copy of our **BIG-TYPE** program **free** of charge. This useful program allows you to load any TI-Artist font (can handle fonts over 200 sectors in size) and type it onto any TI-Artist picture you have. As an added feature, you can even type in color!

You've heard about similar products from the competition that only contain some of the features of Graphics Expander, run slower, or make you work in the dark. Now get the one reviewers are calling "clear," "concise," and "quick." Graphics Expander is available for only $10.

Genial Computerware

P.O. Box 183, Grafton, MA 01519

•To place an order, please send check or money order plus $1 for shipping and handling.

• Credit Card orders (Visa, MC, AmEx) may be placed through Disk Only Software at 1-800-456-9272.

• For a complete catalog of Genial Computerware products for the TI-99/4A and 9640, send a self addressed stamped envelope to the address above.

• Graphics Expander is in Version 2.0. Owners of previous versions may upgrade by returning their original disk and $3.

The Gramulator

GRAM Kracker's successor

By J. PETER HODDIE

Finding a place to start a review of the Gramulator from CaDD Electronics is a difficult task. The Gramulator is the 1988 incarnation of the GRAM emulator, one of the most exciting peripherals developed for the TI99/4A. The Gramulator is the successor to MG's popular GRAM Kracker (reviewed in March 1986 MICROpendium). This review will attempt to cover all the details of the Gramulator while also offering some comparisons to the GRAM Kracker, which is the standard that many will judge it upon.

THE BASIC IDEA

GRAM emulators such as the Gramulator, GRAM Kracker and GRAM Karte allow for a cartridge to be saved on disk and then loaded back into the device for use without the cartridge. Once the cartridge is residing in the emulator it can be modified because it is now contained in RAM memory. Thus a GRAM emulator allows for customization of cartridges. Further, TI BASIC and a large portion of the "operating system" of the 99/4A are contained in GROM memory in the console, which the Gramulator is able to emulate. This allows for changes in the actual behavior of the 99/4A from the moment it is powered up.

GRAM devices allow for a complete backup of a cartridge library to be made to disk. This can provide some degree of security against damage to cartridges. While some might argue that a GRAM emulator encourages piracy of cartridges by allowing them to be freely copied, this is hardly realistic. Most TI users now own every cartridge they are likely to ever want, and the cost of cartridges is generally no more than $10, and often less. With GRAM devices selling for over $150, buying such a device for piracy purposes hardly seems economical.

A GRAM device also reduces the wear on your the console cartridge port. Once placed in the cartridge port, there is usually no reason to remove the device, and thus one of the most frequent causes of problems with the 99/4A (ever had Extended BASIC lock up for no good reason?) is essentially eliminated.

Using a GRAM device a programmer can write programs in GPL (TI's proprietary Graphics Programming Language), which is nearly impossible without a GRAM emulator (although using Monty Schmidt's clever GPL Linker it is possible). There are several GPL assemblers and disassemblers available for those who wish to program the 99/4A in the language that most TI users chose for most of their cartridges.

Review

Report Card

Performance.....A
Ease of Use.....B+
Documentation.....A–
Value.....A
Final Grade.....A

Cost: $185
Manufacturer: CaDD Electronics, 52 Audubon Road, Haverhill, MA 01830
Requirements: 32K expansion, disk system

THE HARDWARE

The Gramulator is a small black metal case, that slides into the cartridge port of the 99/4A console, and extends to fill the entire length and width of the cartridge port. There are two rubber "feet" to hold it level. In front of the cartridge port there is a connector to plug in a cartridge, and the battery which backs up the RAM memory when the console is turned off. The placement of the battery outside the case is a major improvement over the GRAM Kracker in terms of changing the battery. With the GRAM Kracker, this involves completely disassembling the case, as often as every six months. By locating the battery out of the case, even a technoklutz can easily replace the battery in just a few seconds.

The Gramulator contains 96K of RAM configured to emulate the 64K of GROM/GRAM in the console and cartridges, 2 banks of 8K RAM/ROM that may also reside in a cartridge, and 2 extra banks of 8K for later expansion. There is also 8K of ROM which contains the Gramulator software. On the front are 7 two position switches which control the devices operation. These switches control similar functions to those on the GRAM Kracker, with further operational details below.

With instructions available from CaDD, the Gramulator may be modified by the addition of a few chips to utilize the two extra 8K RAM banks to emulate the Milton Bradley MBX cartridges. The Gramulator is the first product available for modifying these unique cartridges.

The Gramulator comes with a three-month warranty on parts and labor.

THE SOFTWARE

Built into the Gramulator is 8K of software which allows for loading and saving cartridges, TI BASIC, and the operating system, in addition to setting many of the device characteristics that are also controlled by the switches. There are also options to save specific ROMs and GRAMs, not necessarily as part of a cartridge. In a cartridge, a GROM contains 6K of data, although it occupies an 8K block of memory. This extra 2K is extremely useful for programmers when modifying or enhancing cartridges, because it is never used in existing cartridges. When saving a complete cartridge the software saves "small GROMs" which take up 8 fewer sectors on disk than "large GROMs" which are saved when the option to save specific memory banks is used. In this way, the casual user doesn't have to waste disk space on unused areas of memory, while a programmer has the ability to save these areas if necessary. The GRAM Kracker software saved in 8K banks to be safe.

The built in software also contains a catalog routine (which handles floppy, RAM, and hard disks) to help locate files. Using the Gramulator system software, a memory editor may be loaded from disk. The memory editor features editing of both GRAM and CPU memory, editing and display in both ASCII and hexadecimal, string search, printer/disk dumps, and the ability to fill blocks of memory with a particular byte value.

Taken together, the system software and memory editor allow for nearly all the capabilities of the GRAM Kracker inter-

(See Page 41)

GRAMULATOR—

(Continued from Page 40)

nal software with some welcome improvements, such as selective memory bank save, disk catalog and small GROM size. The software is not as "pretty" as the GRAM Kracker's and the memory editor resides on disk as opposed to memory, which is not as convenient.

For those who wish to modify the system software or the memory editor, the source code is available for a nominal fee from CaDD. The code is well written, and carefully commented.

Providing the source code for systems software is becoming a popular option within the TI community (witness the success of the Horizon RAMdisk) and CaDD is to be commended for choosing this open policy.

DOCUMENTATION

The manual provided with the Gramulator is entirely adequate. It clearly describes the functions of all the hardware switches, how to operate the system software and memory editor, some specific examples of modifying cartridges, some useful advice on using the device, and an excellent table of contents to make finding information fast. The manual is typeset, printed on 8½ x 11 paper, and contains 36 pages.

Nothing is wrong with the Gramulator manual; unfortunately, it must be compared to the GRAM Kracker manual.

If there was one thing MG did better than anyone else, it was to write outstanding documentation. The GRAM Kracker manual, no exception, is packed with examples, hints, cartridge modifications, and technical and programming information. The GRAM Kracker manual ($2.50 including shipping) is still available from MG, 1475 W. Cypress Ave., San Dimas CA 91773. This manual makes a worthwhile addition to a Gramulator owner's reference library.

SO WHAT DOES IT DO FOR ME?

The Gramulator moves any cartridge from ROM where it can not be edited, into RAM. This allows anyone to make changes in a cartridge. Changes can be simple — such as changing screen colors — to the more complex — such as adding new CALLs to Extended BASIC, or modifying the TI title screen to include your name. Using the supplied memory editor it is easy to make simple modifications.

With the proliferation of large RAMdisks and now hard drives, the ability to store large numbers of cartridges on-line becomes a reality. With these devices, a cartridge may be loaded in just a few seconds. No longer do you have to search for cartridges on your desk. The Gramulator sits in the cartridge port and you simply load the cartridge you need. Unlike the GRAM Kracker, loading a cartridge often requires no manipulation of switches on the device, which can save considerable time.

For those using an external keyboard (such as produced by Rave 99), the Gramulator allows placement of the console completely out of the way since you no longer need to access the cartridge port. For most operations there is no need to manipulate the switches, so the console can be conveniently placed to one side.

SUPPORT

Because of the popularity of the GRAM Kracker, particularly in the programming community, a large number of articles were written explaining how to make changes to cartridges and the operating system. An excellent collection of these is Kracker Facts, edited by Mike Dodd, and available from the L.A. 99'ers. The vast majority of the material in this book is directly applicable to the Gramulator.

Programs are also available in the public domain and fairware circles which take advantage of the Gramulator. There are some particularly good articles written by Marty Kroll available on CompuServe and from several software libraries, including the Boston Computer Society.

A program I wrote called GRAM Packer, available from Genial Computerware, will work with the Gramulator, though at this time not all features are functional. For details see the review in the December 1986 MICROpendium.

CONCLUSION

Most people who purchased a GRAM Kracker wondered how they ever did without it. The Gramulator is a worthy successor to this legacy.

If you have ever considered purchasing a GRAM emulator product, now is the time. The Gramulator is a powerful addition to the 99/4A environment that will change the face of your computer.

Barrage

Cannons against 'acid rain'

By KEITH BERGMAN

If a Hall of Fame existed for TI games, I would nominate DataBioTics' Barrage for entry immediately. The game is fast-paced, well-written and full of enough action to wear out a set of joysticks (not to mention hands — trust me, I know!)

Barrage opens up with a rather nondescript title screen, from which the player chooses one of three options — one player, two player cooperative and two player competitive. This is a nice feature if you, as I do, have a family full of people who like games and hate waiting to play. In both two player options, the two players play simultaneously.

After the options, the scene shifts to the main game screen. At the bottom in each corner sits a laser cannon, atop a platform. Beneath the platform, each cannon's ammunition is stacked up. Between the cannons, on the ground, lie eight objects — cars, planes, satellite dishes and the like — that the player(s) are to defend from the Draks. Who, you ask, are the Draks?

The Draks are the fiendish enemies of this game. However, unlike in most arcade games, we never see the Draks. We see only the acid balls they are raining down upon the planet. In Level 1, a slew of acid balls — actually dots — fall toward the objects below. The player(s) shoot them by positioning an on-screen target on them and firing. A laser beam shoots out from the laser cannon to the target and explodes, blowing up (you hope) the acid ball in the process. This method of shooting is similar to that in the arcade game "Missile Command."

In Level 2, we meet up with a new type of acid ball: a larger, red ball that seems to drift lazily to the bottom of the screen. However, when the red balls get close to an object or a laser cannon, they cluster together and move almost as one to destroy their objectives. After this, in Level 3, an even worse menace appears. I have christened these "fastballs" and they live up to their name! They look just like the first type of acid ball, but they shoot our from the top of the screen, zipping toward their targets. You'll usually get them only by shooting ahead of them and letting them run into your explosion.

When an acid ball comes into contact with a laser cannon or an object on the ground, the ball destroys it. Laser cannons that are destroyed are rebuilt at the end of each level, but objects stay destroyed.

Once all of the objects a player is guarding are destroyed, that player's game is over. A destroyed object is rebuilt at 50,000 points, 100,000 points and at every interval of 100,000 afterwards.

If only one person is playing (a rare occurrence around my house!), the player controls one target. Both laser cannons are at his or her disposal and when he or she shoots, the beam comes from the cannon closest to the target.

When two people play cooperatively, they each control one target and fire from one laser cannon. They work together, and there is one score kept. If two play competitively, the control is the same, but each has his or her own score. Each player defends only four of the eight objects, the four closest to the defender's cannon.

There are several bonuses in Barrage. If, during a level, you destroy three or more acid balls with one explosion, you will get a bonus that varies depending on what level you are on and how many were shot. At the end of each level, you get 10 points times the level number (10*LEVEL in XB) for each shot left unfired below your laser cannon and 300*LEVEL for each object left intact. At this time, between levels, is when objects are rebuilt if one's score is high enough. If all of your objects have been destroyed but you have achieved a high enough score to get an object rebuilt, it will be rebuilt and your game will continue. Otherwise, it's GAME OVER.

Some things to note: never, *NEVER*, let the acid balls destroy your laser cannon. If two persons are playing competitively, each player controls only one, so if it's destroyed, your chances of making it to the next level are very slim.

Also, beware of Level 8. Fastball after fastball will whiz out of the top of the screen. For unexperienced players, this level is often doom. The fastballs are deceptive, too. On some levels, they will come down on one side of the screen, and, while you are trying to shoot them, some slow ones move in on the other side and take out a cannon or some objects.

This game's action is fast and furious. Two players playing simultaneously does not appear to slow the game down at all, even with a wave of acid balls falling as well. Once you get the hang of maneuvering the target, you'll be flying off the screen, shooting and heading for the next acid ball before the explosion goes off. The game is simple enough to dive into without reading the docs, but they're adequate.

Barrage can take its place among the great action games on the 99/4A. It's sure taken its place in Slot 3 of my Widget! I get less and less done all the time! (And I challenge anyone to beat my score of 488,010 on Level 15 — and that was with a pair of TI joysticks!

Review

Report Card

Performance	**A**
Ease of Use	**A**
Documentation	**A**
Value	**A**
Final Grade	**A**

Cost: $19.95
Manufacturer: DataBioTics, P.O. Box 1194, Palos Verdes Estates, CA 90274
Requirements: Console, monitor, joysticks

Reader to Reader

"I have been hunting for a CALL PEEK address for over TWO YEARS, and I still haven't found it! (It's beginning to make me crazy!) Can ANYBODY tell me the address (and "values" concerned) to CALL PEEK from within a running XB program, (AND WITHOUT PRESSING ANY KEYS) to check if the ALPHA LOCK is up or down?! I KNOW it CAN be done, but HOW?! (Mucho thanks!)" writes Ray Kazmer, 13225 Azores Ave., Sylmar, CA 91342

Eunice Spooner, Box 3720 Webb Rd., Waterville, ME 04901, would like to know how to get TI Logo II designs printed on an Epson-compatible printer.

Jim Lohmeyer, 801 East Center, LeRoy, IL 61752, says he loves to program but needs ideas for programs, subroutines, modifications or projects. Any person who submits an idea he uses will be the first to receive a copy of his fairware programs.

Newsbytes

Nova Scotia fair set

TI of Nova Scotia Users Group will hold its first TI Atlantic Fair Oct. 1 in Halifax, Nova Scotia, Canada.

Russ Portolesi, TINS secretary, said group members hope the fair will become an annual event.

For further information, contact TINS, 2846 Gottingen St., Halifax, N.S., Canada B4C 1V5; TEXTNET BBS at (902) 455-2076; or voice at DATA*PORT (902) 454-0232.

RSTS to distribute Myarc products in UK

Richard Sierakowski Technical Services International has become the sole distributor of Myarc products in the United Kingdom.

The firm also provides services to TI99/4A users.

For further information, write RSTS, Old School Buildings, Herd St., Marlborough, Wilthsire, SN8 1DG, England.

Asgard releases new products

Asgard Software has released two new games, Column Attack! and Oliver's Twist; a graphics package, Disk of Dinosaurs; an Extended BASIC utility, Quick-Run; and a RAMdisk manager, RAM*Boot.

Column Attack!, an arcade-type game, is an attack on earth by the Flugalins of the planet Flugal, who plan to level everything in sight and construct fast-food restaurants, shopping malls and concrete parking lots. The game user control's earth's only laser defense gun and tries to destroy the attacking aliens before they can set up their laser stations or, barring that, destroy the stations.

Column Attack! by Chris Bobbitt is written in Fortran 99. It requires TI Extended BASIC, Editor/Assembler or TI-Writer, 32K and one disk drive. Suggested retail is $9.95.

Oliver's Twist is an adventure game in which the user will inherit the castle of his recently deceased Uncle Oliver (King Oliver IV) on condition that the user mollifies his ancestral spirits by returning 15 treasures to their rightful places. The user also has to figure out where the rightful places are without disturbing Oliver's ghost.

The game was written by Mickey Schmitt and Lynn Gardner and was tested by members of the Pittsburgh Users Group. It requires the Adventure module. Suggested retail is $9.95.

Disk of Dinosaurs, by Ken Gilliland, is a two-disk package of dinosaur graphics stored in TI-Artist Instance format, including dinosaur pictures; a complete dinosaur alphabet designed to let the user "dinosaurize" letters, reports and signs; background scenes; a "dinosaur hunting license"; and four dinosaur cartoon shorts.

It requires TI-Artist or a program that can use TI-Artist work, 32K memory expansion and a disk system (either a CorComp or TI controller to view the cartoons). A printer is recommended. Suggested retail is $12.95.

Quick-Run is said to allow the user to shave minutes off the start-up time of any Extended BASIC program, including those with assembly language, by taking a "snapshot" of a running XBASIC program and saving it to disk with the scren setup and all variables. Later, according to the manufacturer, the user can run the saved copy and the program picks up where it left off, so the user does not have to wait for initialization, for instance.

Asgard says Quick-Run allows the user to chain programs together and store the variables in memory instead of writing them to disk and also lets the user run a program from any line number.

Quick-Run was written by Travis Watford and requires XBASIC, 32K and a disk system. Suggested retail is $9.95.

RAM*Boot is called the only RAMdisk manager designed specifically for the Myarc 128K and 512K cards.

The manufacturer says the user can have it set up, copy and number of disks to it and then run a program from the RAMdisk card. Once the RAMdisk is initialized, according to Asgard, it becomes a RAMdisk manager that allows the user to back it up to floppy disk, copy disks or run any program.

The program is by Watford. It requires XBASIC, 32K and one disk drive. Suggested retail is $9.95.

For further information or to order, contact Asgard Software, P.O. Box 10306, Rockville, MD 20850.

Comic Show Editor won't run on 9640

Ray Kazmer, who wrote an article on how to use the Comic Show Editor in the July 1988 MICROpendium, has notified us that the program will not run on the Geneve 9640.

East Texas group to participate in fair

The Longview Computer Users Group will again have the use of a double booth at the Gregg County (Texas) Fair. This year's fair will run Sept. 12-18.

Leo DuBry of the group said the TI99/4A Sig will have approximately one-fourth of the booth.

99 BBS adds support for 9640 online

The 99BBS in Whittier, California, has recently added support for the Geneve 9640, according to Roger Davis, sysop.

The 24-hour, seven-day-a-week board operates at from 300 to 2400 baud and is PC Pursuit accessible (CALAN). Phone number for the board is (213) 947-7777.

This board has been in operation as versions 5.0 through 7.6 for more than three years, Davis says. He says there is no wait for verification for users, but full accessability on the first call.

Minimum equipment requirements for the board are a TI console, Extended BASIC, RS232, monitor, memory expansion, disk drive and controller and a smart modem (preferably fully Hayes compatible), Davis says. He says he highly recommends multiple disk drives. Other optional equipment includes a clock card (setup is for Corcomp, but any may be used), a RAM disk and a printer.

The board is available by sending a DS/DD disk, or equivalent, return postage and a note requesting the board to Roger Davis, 11410 Grovedale, Whittier, DA 90604. Davis urges current 99BBS system sysops to send a disk for the update.

(See Page 44)

User Notes

Editor Aid helps XBASIC programmers

If you've ever wanted to save or delete a block of program lines in an Extended BASIC program without having to manually delete individual lines, then Editor Aid is for you. The program allows programmers to save or delete designated blocks of code. Users may also produce a cross-reference listing of line numbers that may be output to the screen or a designated device.

The program was written by Merv Kroll, a member of the Brisbane (Australia) TI User Group. It appeared in the group's newsletter.

To use Editor Aid, first save the program you wish to act on in MERGE format. Then follow on-screen instructions and prompts. After performing deletions or excisions, the program saves the result in MERGE format.

```
100 !*********************
!031
110 !*      EDITOR AID      *
!204
120 !*********************
!031
130 CALL CLEAR !209
140 DISPLAY AT(1,3):RPT$("*"
,22)!2,3,4 FOR PROGRAM NAME
!114
150 DISPLAY AT(3,3):"*
 EDITORAID";TAB(24);"*" !201
160 DISPLAY AT(5,3):RPT$("*"
,22):: DISPLAY AT(7,3):"AUTH
OR:" !066
170 DISPLAY AT(13,9):"MERV K
ROLL";TAB(7);"23 MUSGRAVE ST
";TAB(6);"KIPPA-RING 4020" !
000
180 DISPLAY AT(24,2):"MEMBER
 BRISBANE USER GROUP" :: FOR
 I=1 TO 500 :: NEXT I !030
190 CALL CLEAR :: PRINT "THI
S PROGRAM WILL ENABLE YOU":"
TO SAVE A SECTION OF A ":"PR
OGRAM, DELETE A BLOCK" !101
200 PRINT "OF LINE NUMBERS W
ITH EASE,":"OR PRINT A LINE
NUMBER CROSS":"REFERENCE." :
193
210 PRINT :: PRINT "FIRST, S
AVE THE PROGRAM AS A ":"`MER
GED' FILE, THEN AS" !075
220 PRINT "ASKED, ENTER PROG
RAM NAME,":"NAME YOU WISH TO
 RE-SAVE AS,":"AND BOTH LINE
 NUMBERS." !159
230 PRINT :: PRINT "LINE NUM
BERS YOU ENTER ARE":"BOTH IN
CLUSIVE....." :: PRINT ::
 PRINT :: PRINT !013
240 INPUT "PRESS <ENTER> TO
CONTINUE.":ANS$ !073
250 CALL CLEAR :: DISPLAY AT
(3,3):"DO YOU WISH TO:-" ::
DISPLAY AT(8,1):"(1) SAVE A
PROGRAM PORTION" !056
260 DISPLAY AT(10,1):"(2) DE
LETE A PORTION" :: DISPLAY A
```

(See Page 45)

Newsbytes

(Continued from Page 43)

Pennsylvania users set exhibition

The Central PA 99/4A Users Group is sponsoring its 1988 Computer Exposition from 7 a.m. to 2 p.m. Oct. 16 at the Carlisle Fair Grounds in Carlisle, Pennsylvania.

The exposition is in conjunction with the annual Hamfest of the Cumberland County Amateur Radio Society.

Admission to the event is free, according to Barry Long, secretary of the user group; however, there is a gate charge to the fairgrounds.

Tables are available to vendors and suppliers. An eight-foot table is $40, and a 16-foot space (two tables) is $50, Long said. A 23-inch color monitor will be available for demonstrations of products in a demonstration area. Reservation deadline for vendors is Sept. 23.

To reserve vendor space, write the Central PA 99/4A Users Group, P.O. Box 14126, Harrisburg, PA 17104-0126. For more information contact Long after 8 p.m. at (717) 564-2975 or contact the group's president, Dave Ratclife, at (717) 238-5414.

6th Chicago TI Faire is Nov. 12

The Chicago TI Users' Group's 6th Annual TI Faire will be held from 9 a.m. to 6 p.m. Nov. 12 at the Holiday Inn in Rolling Meadows, Illinois.

Admission is $4. Optional events are a social mixer from 8 p.m. to midnight Nov. 11, for $4, and a dinner at 7 p.m. Nov. 12 for $10.

For further information contact the Chicago TI Users Group, P.O. Box 578341, Chicago, IL 60657 or call (312) 755-0051.

User Notes

(Continued from Page 44)

```
T(12,1):"(3) CROSS REF. LINE
 NUMBERS" :: DISPLAY AT(14,5
):"YOUR CHOICE" !058
270 ACCEPT AT(14,17)BEEP VAL
IDATE("123"):A !196
280 ON A GOTO 290,520,740 !0
54
290 REM SAVE PORTION !052
300 CALL CLEAR !209
310 PRINT "READ FROM DSK1."
:: ACCEPT AT(23,14)SIZE(-12)
BEEP:READ$ !001
320 PRINT :"SAVE TO DSK2." :
: ACCEPT AT(23,12)SIZE(-12)B
EEP:WRITE$ !164
330 INPUT "LOWER LINE NUMBER
    ":LOWER !230
340 INPUT "UPPER LINE NUMBER
    ":UPPER !236
350 OPEN #1:"DSK"&READ$,DISP
LAY ,INPUT ,VARIABLE 163 !14
8
360 OPEN #2:"DSK"&WRITE$,DIS
PLAY ,OUTPUT,VARIABLE 163 !1
05
370 EOF$=CHR$(255)&CHR$(255)
!200
380 LINPUT #1:X$ !210
390 IF SEG$(X$,1,2)=EOF$ THE
N 480 !182
400 LINE1=ASC(SEG$(X$,1,1))!
123
410 LINE2=ASC(SEG$(X$,2,1))!
125
420 LINENUM=LINE1*256+LINE2
!117
430 IF LINENUM<LOWER THEN 47
0 !052
440 IF LINENUM>UPPER THEN 48
0 !066
450 PRINT #2:X$ !197
460 PRINT "LINE";LINENUM;"SA
VED" !078
470 GOTO 380 !204
480 PRINT #2:CHR$(255)&CHR$(
255)!085
490 CLOSE #1 !151
500 CLOSE #2 !152
510 STOP !152
520 REM DELETE SECTION !162
530 CALL CLEAR !209
540 PRINT "READ FROM DSK1."
:: ACCEPT AT(23,14)SIZE(-12)
BEEP:READ$ !001
550 PRINT :"SAVE TO DSK2." :
: ACCEPT AT(23,12)SIZE(-12)B
EEP:WRITE$ !164
560 INPUT "LOWER LINE NUMBER
    ":LOWER !164
570 INPUT "HIGHER LINE NUMBE
R ":UPPER !181
580 OPEN #1:"DSK"&READ$,DISP
LAY ,INPUT ,VARIABLE 163 !14
8
590 OPEN #2:"DSK"&WRITE$,DIS
PLAY ,INPUT ,VARIABLE 163 !0
04
600 EOF$=CHR$(255)&CHR$(255)
!200
610 LINPUT #1:X$ !210
620 IF SEG$(X$,1,2)=EOF$ THE
N 700 !147
630 LINE 1=ASC(SEG$(X$,1,1))
!068
640 LINE 2=ASC(SEG$(X$,2,1))
!070
650 LINENUM=LIN1*256+LINE2 !
048
660 IF LINENUM<LOWER OR LINE
NUM>UPPER THEN 670 ELSE 680
!015
670 PRINT #2:X$ :: GOTO 690
!074
680 PRINT "LINE ";LINENUM;"D
ELETED" !245
690 GOTO 610 !179
700 PRINT #2:CHR$(255)&CHR$(
255)!085
710 CLOSE #1 !151
720 CLOSE #2 !152
730 STOP !152
740 PRINT "READ FROM DSK1."
:: ACCEPT AT(23,14)SIZE(-12)
BEEP:READ$ !001
750 DISPLAY AT(8,1)ERASE ALL
:"PRINT TO (S)CREEN OR (O)TH
ER":"          ?" :: ACCEPT
AT(9,12)BEEP VALIDATE("SO"):
ANS$ !063
760 IF ANS$="S" THEN 790 ELS
E DISPLAY AT(11,1):"DEVICE N
AME?" !006
770 ACCEPT AT(11,14)BEEP:ANS
$ !184
780 OPEN #2:ANS$,OUTPUT !252
790 IF ANS$<>"S" THEN ANS=2
ELSE ANS=0 !137
800 OPEN #1:"DSK"&READ$,DISP
LAY ,INPUT ,VARIABLE 163 !14
8
810 PRINT #ANS:"LINE CROSS R
EFERENCE OF    ";PROG$: : :!
247
820 EOF$=CHR$(255)&CHR$(255)
```

(See Page 46)

User Notes

(Continued from Page 45)

```
!200
830 LINPUT #1:X$ !210
840 IF SEG$(X$,1,2)=EOF$ THE
N CLOSE #1 :: STOP !189
850 LINE1=ASC(SEG$(X$,1,1))!
123
860 LINE2=ASC(SEG$(X$,2,1))!
125
870 LINENUM=LINE1*256+LINE2
!117
880 FOR I=3 TO LEN(X$)-3 ::
COUNT=I !206
890 B=ASC(SEG$(X$,I,1))!179
900 IF B=201 THEN GOSUB 930
!135
910 NEXT I !223
920 GOTO 830 !144
930 BYTE3=ASC(SEG$(X$,COUNT+
1,1))!211
940 BYTE4=ASC(SEG$(X$,COUNT+
2,1))!213
950 PRINT #ANS:LINE1*256+LIN
E2,BYTE3*256+BYTE4 !061
960 RETURN !136
```

POKEV and Super Extended BASIC

Arthur Dubeau, of Woonsocket, Rhode Island, writes:

I may have something to contribute to others who had the same problem but haven't found a solution. It is limited to those who have Super Extended BASIC. If don't know if it will work with others versions of Extended BASIC, but I know it doesn't work with TI Extended BASIC.

You published two programs — VDPUTIL 2 & 3 — to change BASIC to XBASIC. With SXB the problem is in the CALL LINK("POKEV",VAR) statement. Everytime it came up my computer would either lock up or I'd get a "subprogram not found" error. Since SXB has a CALL POKEV(VAR) I use that in place but with the same variables in the parenthesis. Voila! It worked. In other words, I put the POKEV outside and eliminated "Link."

Color selector

This comes from Edwin G. Donovan, of Monroe, Washington. He writes:

The attached program listing will display all 512 color selections on the Geneve computer if Mike Dodd's program Color Selector (May, 1988) is on DSK1 with a filename of COLOR. Run this Extended BASIC program and follow display directions. The screen color will change with each key press of R, G or B.

```
10 !Use with MIKE DODD'S Col
or Selector Program-May 1988
MICROpendium & TI-Ext BASIC
!156
20 CALL INIT :: CALL LOAD("D
SK1.COLOR"):: CALL CLEAR !24
9
30 CALL SCREEN(2):: DISPLAY
AT(12,11):"GENEVE'S" !216
40 DISPLAY AT(14,5):"COLOR D
EMONSTRATION" :: DISPLAY :"K
ey R-G-B for next color":"or
 press Q to Quit program" !1
94
50 R,G,B=7 :: DISPLAY AT(4,3
):"R  G  B","    Diggin's" !
252
60 CALL LINK("COLOR",1,R,G,B
):: DISPLAY AT(2,2):R;G;B,"
   Donovan's" :: CALL SCREEN
(1):: CALL KEY(0,K,S)!217
70 IF K=71 THEN 130 !29 !168
80 IF K=82 THEN 110 !July !2
07
90 IF K=66 THEN 150 !1988 !0
47
100 IF K=81 THEN END ELSE 60
 !129
110 R=R+1 :: IF R=8 THEN R=0
 !237
120 GOTO 60 !Donovan's !125
130 G=G+1 :: IF G=8 THEN G=0
 !193
140 GOTO 60 !Diggin's !250
150 B=B+1 :: IF B=8 THEN B=0
 !173
160 GOTO 60 !415 Ferry St.
   Monroe, WA 98272 !005
```

Slash zero for 80-col. Multiplan

Here's a fix from Myarc for those who use Multiplan. The following code permanently replaces the zero character with a slash zero. It is designed for the 80-column version of Multiplan that runs on the Geneve.

Using a sector editor on the MPCHAR file, locate the following code: >0018 2424 2424 2418. Replace it with the following: >FFE7 DBDB DBDB D BE7.

This will change the standard zero character to a slash zero. Now for the inverse video zero. Locate the following code: >0030 4858 7868 4830. Replace it with the following: >FFC7 B7A7 8797 B7CF.

Print-Trace outputs to printer

This routine comes from TISHUG in Australia. Written by Mike Slattery, the program is used to output a CALL TRACE readout to a printer. The routine should be merged into the program you want to trace prior to running.

```
9100 OPEN #1:"PIO" !253
9110 PR$="" !075
9120 FOR R=1 TO 24 :: FOR C=
3 TO 28 :: CALL GCHAR(R,C,X)
:: IF X=60 THEN 9140 :: IF X
=31 OR X=32 THEN 9150 :: IF
X=62 THEN X=32 !250
9130 PR$=PR$&CHR$(X):: CT=CT
+1 :: IF CT>75 AND(X=32 OR X
=31)THEN PRINT #1:PR$ :: PR$
="" :: CT=0 !031
9140 NEXT C !217
9150 NEXT R !232
9160 PRINT #1:PR$ :: PR$=""
:: CT=0 !171
9170 CLOSE #1 :: CALL CLEAR
!234
9180 RETURN !136
```

User Notes **is a column of tips and ideas designed to help readers put their computers to better use. The information provided comes from many sources, including TI user group newsletters and readers. MICROpendium pays $10 for any item sent in by readers that appears in this column. Mail User Notes to: MICROpendium User Notes, P.O. Box 1343, Round Rock, TX 78680.**

Coming up

- Review of Myarc's hard and floppy disk controller
- Stephen Shaw on the new Turbo Pascal

Classified

Software

CHECKBOOK RECONCILE $29.95
Balance to the penny every month. Rocketman Checkbook Program. See our program review in August 1987 MICROpendium. California Programs, 4426 Appian Way, El Sobrante, Ca. 94803. 415-222-1626. v5n9

HOME BANKING
Common sense, productivity software for your 99/4A. For more information, write to E & M Software, Box 551, Oscoda, MI 48750. v5n11

SLIDEMAKER
ARTIST UTILITY DRAWS SLIDES AND INSTANCES.

PROGRAFIX
DRAW GRAPHICS FOR PROGRAM USE ON SCREEN OR PRINTER.

TEXTGRAFIX
DRAWS GRAPHICS FOR TEXT
NO .TL'S ALL $12.00

JIM UZZELL
6659 CANYONWAY
HOUSTON, TX. 77086

COPYRIGHT 1988 by DDI v5n7

GENIAL COMPUTERWARE
Credit Card orders for Genial Computerware products. Visa, M/C, Amex. Call 1-800-456-9272 or write DOS Software Division, P.O. Box 244, Lorton, VA 22079. v5n7

ASSEMBLY LANGUAGE MUSIC
Assembly Language Music that plays under Extended BASIC. 20 pieces by Johann Sebastian Bach from Anna Magdalena's Notebook. Features a selection menu Load program, and makes your TI sound like a Grand Piano. Requires XB, Disk, 32K. Send $4.50 to Harrison Software, 5705 40th Place, Hyattsville, MD 20781. v5n7

Hardware

HORIZON RAM DISKS
Credit Card orders for Horizon RAM Disk products. Visa, M/C, Amex. Call 1-800-456-9272 or write DOS Software Division, P.O. Box 244, Lorton, VA 22079. v5n7

Systems

SEVERAL EXPANDED 99/4A SYSTEMS
Console, P-box, 32K, RS232, SS/SD, DS/DD. Steve 316-685-2096 Days, Curtis 417-869-3802 Days. v5n8

FOR SALE
TI99/4A Computer, Color Monitor, Expansion Box, Disk Drive, Controller, RS 232, Modem, 32K Expansion Card, Voice Synthesizer, Extended BASIC, Editor/Assembler, TI-Writer, Donkey Kong, Centipede, Much More, $550.00. Call Mark 713-556-6428. v5n7

Policy

The cost of classified advertising is 25 cents per word. Classified display (i.e., special formatting or graphics) is $9 per column inch. Classified advertisements must be paid in advance. Classified advertisers may request a category under which they would like their advertisement to appear, but the final placement decision is the responsibility of the publisher.

Classified deadlines will be kept open for as long as practical. For the purpose of classified advertising deadlines, any classified ad received later than the first day of any month cannot be assured of placement in the next edition. We will do our best to include every advertisement that is submitted in the earliest possible edition.

The publisher offers no guarantee that any advertisement will be published in any particular issue. Any damages that result either from errors in copy or for failure to be included in any particular edition will be limited to the amount of the cost of the advertisement itself. The publisher reserves the right to reject any advertisement.

The advertiser may elect to publish the advertisement in subsequent editions at the same charge, payable prior to publication. The deadline for carryover classifieds is the same as for new advertising.

In submitting an ad, please indicate whether you would like a refund if it is not published in the requested edition or whether you would like us to hold it for the next edition. Cancellations and refunds cannot be made after the second day of the month.

Send classified advertising to: MICROpendium, P.O. Box 1343, Round Rock, TX 78680.

Miscellaneous

99/4A P-CODE CARD WITH ALL DISKS AND MANUALS	$140
P-CODE CARD ALONE	$ 89
PE-BOX/RS232/32K/DRIVE/CONTLR	$349
18' P-BOX FLAT EXTENSION CABLE	$ 22
PARALLEL PRINTER CABLE 6'	$ 20
WORD WRITER +	$ 45
STAND ALONE DISK DRIVE (NEW)	$ 89
STAND ALONE DISK DRIVE DSDD	$ 99
TI ORIGINAL 32K MEMORY CARD	$ 79
TI ORIGINAL COLOR MONITOR	$175
IBM THERMAL PRINTER FOR TI	$ 60
AVATEX MODEM 1200HC	$ 99
AVATAX MODEM 1200e	$ 89
PLUS 10% SHIPPING	CALL OR WRITE

JIM LESHER, 722 HUNTLEY
DALLAS, TX. 75214, 214 821 9274 v5.n7

MICROpendium magazine holders are in!

$3 per set of 12. (Texas residents add 7.5% sales tax.) Send check or money order to **Holders, MICROpendium, P.O. Box 1343, Round Rock, TX 78680.**

Texas Instruments
TI99/4A
Hardware & Software

— TURBO-PASC '99 in stock! only $59.95 — Proto-Board for PEB by Willforth only $35. TI Dual Sticks $10.00 XB Detective $10.00 Speech Editor $20. Printer Cable (PIO) $24. Jungle Hunt, Protector II, Moon Patrol, $15 ea. Dust Cover $4, Q'bert $15, Adventure $10.00 Hint Bk $7.95, Super Extended Basic $59.95 ** Sorcerer by Infocom (disk Dr. req.) $15. P-Code Card w/Software $159. TI-Artist $20. TPA w/Font 1 $34, TI-BASE $24.95 latest Ver Side-Port & Cart-port Ext. Cable $25.00 ea. We stock 32K, Disk Ctrl, RS232, Ramdisk Cards

L.L. CONNER ENTERPRISE
COMPUTER&ELECTRONICS
1521 Ferry Street
Lafayette, IN 47904
CALL 317-742-8146 or CALL 317-423-4879
Visa or M/C Accepted, add $3 shipping v5n9

The LEADING monthly devoted to the TI99/4A

Subscription Fees

$20 for 12 issues via domestic second class mail
$25.25 (U.S. funds) Canadian or Mexican delivery ($27.50 for Canada after Sept. 1, 1988)
$23.50 (U.S. funds) for 12 issues foreign delivery via surface mail ($25 after Sept. 1, 1988)
$37.00 (U.S. funds) for 12 issues foreign delivery via air mail
Outside U.S., pay via postal or international money order; personal checks from non-U.S. banks will be returned

Address Changes

Subscribers who move may have the delivery of their most recent issue(s) delayed unless MICROpendium is notified six weeks in advance of address changes. Please include your old address as it appears on your mailing label when making an address change.

Back Issue Policy

Back issues of MICROpendium are available to subcribers only. Those wishing back issues may notify us of the issue(s) desired and include $2.00 per issue desired in a check or money order. No shipping charge in U.S., Canada and Mexico; Texas residents add 7.5% sales tax. For foreign delivery, add 50 cents per issue surface mail, $2 per issue air mail. No discounts on orders of sets. All prices U.S. funds. **OUT OF STOCK: Vol 1, nos. 1-2**

Tell us about it

Please let us what columns or features you like the most about MICROpendium. Rank your selections in order of preference using this form. Return it to us when you renew your subscription.

1. _____
2. _____
3. _____
4. _____

Other suggestions: _____

_____

_____

Send me the next 12 issues of MICROpendium. I am enclosing $ in a check or money order in U.S. funds. Mail to: MICROpendium, P.O. Box 1343, Round Rock, TX 78680

Name _____

Address _____

City _____

State____**ZIP** _____

The set of numbers on the left of your mailing label indicates the cover date of your last issue.

v5,n7